LA RIVISTA DI ELETTRONICA APPLICATA, TECNOLOGIE E PRODOTTI

Fare ELETTRONICA

N° 231 - SETTEMBRE 2004 - ANNO 20

€ 5,50 - Frs 8,50

VITAMINA C

NOTEPAD

CD-ROM IN ALLEGATO

MICROCHIP'S® TECHNICAL LIBRARY

Il progetto completo di un

VIDEO CONTROLLER con MOTION DETECTOR

FEBOT

Il robot di Fare Elettronica

PRIMA PARTE: Le specifiche del progetto

UNIREADER

Progetto di un lettore di Smartcard

Poste Italiane Spa - Spedizione in abbonamento Postale - D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano. In caso di mancato recapito restituire all'editore che si impegna a pagare la relativa tassa presso il CPM di Roserio - Milano

ISSN 1591-2272

INWARE EDIZIONI

www.farelettronica.com

editoriale

L'IMPORTANZA DI UN LED...

Verso la fine degli anni '80 lavoravo come progettista, i miei cavalli di battaglia erano: NE555, LM324, CD4011 e tutta la componentistica discreta che allora il mercato offriva. Man mano che le schede si arricchivano di funzioni, diventavano sempre più complesse circuitalmente e quindi avevo cercato di approfondire la programmazione dei microcontrollori per rimpiazzare gran parte della componentistica discreta; ma i primi approcci si rivelarono un insuccesso. Avevo provato con diversi microcontrollori, ma mi ero sempre arenato di fronte ai manuali da studiare e soprattutto di fronte alla carenza di testi validi sull'argomento.
Poi, un giorno, un mio caro amico mi telefona e in tono euforico mi dice: "ho provato dei nuovi microcontrollori, semplici da programmare e con tante caratteristiche davvero interessanti, si chiamano PIC, puoi telefonare alla XZY e chiedergli di inviarti qualche campione ed un programmatore".
Non so spiegarmi il motivo, forse l'entusiasmo del mio amico o forse il fatto che lui ci era riuscito ed io no, ma ho messo da parte i miei insuccessi nel campo e ho chiamato il distributore. Dopo una settimana avevo in ufficio 10 campioni del PIC16C54 e un programmatore PROMATE, entrambi naturalmente prodotti dalla Microchip, allora pressoché sconosciuta. Mi sono subito messo all'opera ma, anche questa volta, dopo due giorni di tentativi (la mia conoscenza dei microcontrollori era allora piuttosto scarsa) ho abbandonato tutto sullo scaffale e sono tornato ai miei NE555 ed all'allegra compagnia dei componenti discreti.
Qualche tempo dopo ho rincontrato il mio amico e lui, dopo avermi apostrofato per aver abbandonato così velocemente la programmazione dei PIC e per la grossa mole di scuse che utilizzavo per giustificare l'abbandono, mi disse una frase che mi diede da pensare: "… ma perché ti complichi la vita, inizia con l'accendere un led, il resto verrà da se".
La sera stessa sono tornato in ufficio a tarda ora e mi sono seduto alla scrivania determinato a seguire il suo consiglio.
Ho preso il primo esempio che ho trovato sul data-sheet, l'ho modificato e, dopo alcuni tentativi e un paio di ore, una bellissima luce rossa ha illuminato il mio led! Un programma composto da due righe di codice aveva cancellato in un attimo la mia frustrazione.
A questo punto non mi rimaneva che spegnerlo, farlo lampeggiare, legarne l'accensione alla pressione di un pulsante… era fatta, alle prime luci dell'alba i PIC non avevano più segreti per me! Sono andato a dormire consapevole che da quel momento in poi il mio lavoro sarebbe cambiato.
Vi ho raccontato questa mia esperienza, nella speranza che motivi quelli di voi che come me hanno cercato di utilizzare i microcontrollori senza successo. Non iniziate con progetti complessi che vi farebbero soltanto perdere tempo e pazienza, armatevi di un editor, il compilatore ed un semplice programmatore, vedrete che non è difficile, ci vuole solo un po' d'impegno. Cominciate con l'accendere un led, vi ritroverete a scrivere su un display in men che non si dica, il passo è breve e la soddisfazione tanta.
Fare Elettronica ha dedicato ampio spazio ai progetti realizzati con microcontrollori PIC, abbiamo presentato un programmatore di semplice realizzazione (222), una scheda per esperimenti (229/230) ed un corso (da 220 al 228) – *PIC Microcontroller By Example* – attualmente disponibile su CD-ROM. Avete quindi tutti gli strumenti, non vi resta che accendere un led…
Per completare il quadro, questo mese, Fare Elettronica è in edicola con un CD-ROM allegato: *Microchip's Technical Library* . Un tassello importante da aggiungere alla vostra libreria tecnica, che vi risparmierà tanto tempo prezioso speso nella ricerca di documentazione.

Ed è proprio un microcontrollore PIC quello che equipaggia il progetto centrale di questo mese, un *Video-Controller con motion detector*, un progetto professionale realizzato da *Angelo Brustia* che siamo sicuri vi darà molte soddisfazioni. Seguono le nuove puntate dei corsi sull'optoelettronica, sulle smart-card, sullo sviluppo di CPLD, sul bus 1-Wire, le applicazioni embedded, gli stabilizzatori di tensione e naturalmente *Vitamina C* che, a partire da questo numero, sarà seguita da *Antonio Di Stefano*. Infine, oltre alla seconda parte del *Pannello luminoso a matrice di led*, arriva la tanto attesa prima parte del nuovo progetto di Fare Elettronica: *FEBOT*. In questa prima parte vi illustriamo le specifiche del progetto, ma vi annunciamo che molto altro bolle in pentola, quindi non perdetevi i prossimi numeri!

Come ogni mese concludo augurandovi una piacevole lettura e dandovi appuntamento in edicola ad Ottobre con un nuovo e ricco numero di Fare Elettronica.

Tiziano Galizia
t.galizia@farelettronica.com

www.farelettronica.com

DIRETTORE RESPONSABILE:
GianCarmelo Moroni

DIRETTORE DI REDAZIONE:
Tiziano Galizia (t.galizia@farelettronica.com)

PROGETTO GRAFICO E IMPAGINAZIONE:
Graficonsult - Milano (info@graficonsult.com)

HANNO COLLABORATO:
Maurizio Del Corso, Gianroberto Negri, Luca Calore, Luca Tondi, Nico Grilloni, Nicola Ulivieri, Antonio Di Stefano, Angelo Brustia, Andrea Perilli, Giuseppe Modugno, Agostino Rolando, Marco Fabbri,

DIREZIONE - REDAZIONE - PUBBLICITÁ
INWARE srl
Via Cadorna, 27/31 - 20032 Cormano (MI)
Tel. 02.66504794 - 02.66504755 - Fax 02.66508225
info@inware.it - www.inwaredizioni.it

STAMPA:
ROTO 2000
Via Leonardo da Vinci, 18/20 - 20080 Casarile (MI)

DISTRIBUZIONE:
Parrini & C. S.p.a.
Viale Forlanini, 23 - 20134 Milano.
Il periodico Fare Elettronica è in attesa del numero di iscrizione al ROC

UFFICIO ABBONAMENTI
PARRINI & C. S.p.a. Servizio abbonamenti
Viale Forlanini, 23 - 20134 Milano
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
Telefono: 02.66504794 - Fax: 02.66508225
Email: abbonamenti@farelettronica.com

Poste Italiane Spa - Spedizione in abbonamento Postale - D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano
Abbonamento per l'Italia: € 39,00
Abbonamento per l'estero: € 99,00
Per la sottoscrizione degli abbonamenti, utilizzare il modulo stampato in ultima pagina.
Gli arretrati potranno essere richiesti, per iscritto, al seguente costo:
Numero singolo: € 7,50
Numero doppio: € 9,00

Autorizzazione alla pubblicazione del Tribunale di Milano n. 647 del 17/11/2003 INWARE srl.

Richieste di assistenza

Per richiedere assistenza o chiarimenti sugli articoli pubblicati, vi preghiamo di contattare l'autore, il cui nome ed indirizzo email è sempre riportato sotto il titolo dell'articolo stesso.
Nel caso ciò non fosse possibile potete scrivere a mailbox@farelettronica.com, ricordandovi di specificare il numero della rivista ed il titolo dell'articolo per il quale chiedete chiarimenti, oltre al vostro nome, cognome ed indirizzo email.
Tutte le richieste con informazioni insufficienti o anonime non saranno prese in considerazione.

Collaborare con Fare Elettronica

La redazione di Fare Elettronica è alla ricerca di collaboratori per la stesura di articoli, progetti, tutorials, rubriche e libri.
Le richieste di collaborazione vanno indirizzate a Tiziano Galizia (t.galizia@farelettronica.com) e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

Come contattarci

Indirizzo email della Redazione:
redazione@farelettronica.com
Indirizzo email dell'Ufficio Abbonamenti:
abbonamenti@farelettronica.com
I nostri numeri telefonici:
Telefono 02.66504794 Fax 02.66508225
Il nostro indirizzo postale:
INWARE Edizioni
Via Cadorna, 27/31
20032 Cormano (MI)

Elenco inserzionisti

SOMMARIO

hardware

praticamente

tutorial

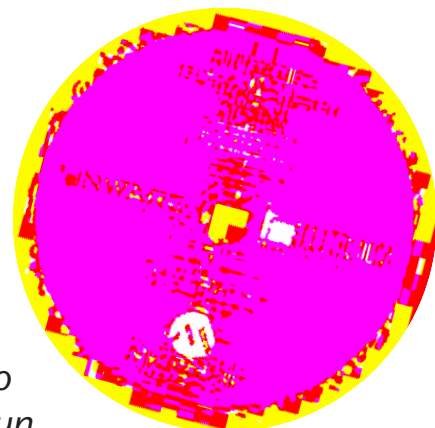
tecnologie sperimentali

robomania

rubriche

CD allegato

MICROCHIP'S TECHNICAL LIBRARY

COME UTILIZZARE IL CD ALLEGATO ALLA RIVISTA

Come annunciato già da diverso tempo, questo numero di Fare Elettronica esce in edicola con un allegato di tutto rispetto. Infatti, grazie all'accordo siglato con la Microchip, possiamo offrirvi in esclusiva il Microchip's Technical Library, un CD che raccoglie la documentazione ed il software per tutti i componenti di questo importante produttore

Inutile dire che la quasi totalità dei progetti presentati nelle riviste di elettronica o che si trovano gratuitamente su Internet, utilizzano un microcontrollore PICmicro®. Questi piccoli mostri hanno conquistato il mercato dei microcontrollori facendo della Microchip, il primo produttore al mondo di microcontrollori a 8 bit.

Ma forse non tutti sanno che la Microchip produce altre linee di componenti, come memorie ed analogici, ma soprattutto che rende disponibile una libreria di Application Notes enorme che offre una soluzione ufficiale a quasi tutte le necessità di programmazione.

Il Microchip's Technical Library contiene una grande quantità di documentazione e software, divisa per categorie, di facile consultazione, inoltre fornisce le ultime versioni di software per tutti i sistemi di sviluppo, prodotti da questa casa.

PREREQUISITI

Prima di poter accedere ai contenuti del CD, vediamo di cosa abbiamo bisogno per farlo funzionare. Il cd si naviga come un sito Web, ma per ovviare alle grosse limitazioni nella dinamicità dei contenuti tipiche dell'HTML, la Microchip ha deciso di adottare Java, quindi dobbiamo assicurarci che il nostro computer disponga dei requisiti minimi.

Inserendo il CD nel lettore, dovrebbe automaticamente aprirsi il browser installato nel nostro computer, sia esso Internet Explorer che Mozilla o Opera. Se il PC rispetta i requisiti dovreste trovarvi di fronte la home page del CD, diversamente, potrebbe non accadere nulla o peggio potremmo ricevere un messaggio di errore tipo "StartHere.exe – Unable to locate DLL". Niente paura, significa che non è stata trovata la Java Virtual Machine, vediamo quindi come installarla.

Figura 1

Aprite il cd da "Risorse del computer", ciccandoci sopra con il tasto destro del mouse e scegliendo la voce "Apri". Vi sarà mostrato l'elenco delle cartelle disponi-

Figura 2

bili. Aprite la cartella "JVM" ed eseguite il file "JavaVM-1.4.2.04.exe" (l'estensione .EXE potrebbe non essere visualizzata su alcuni computer), partirà l'installazione della Java Virtual Machine.
Vi troverete nella situazione di figura 1, cliccate sui tasti contrassegnati dal cerchietto rosso per procedere. Al termine dell'installazione potrebbe essere necessario riavviare il computer. Prima di dare conferma al riavvio, ricordatevi di rimuovere il CD dal lettore.

Dopo aver riavviato il computer, lanciate Internet Explorer, cliccate sul menù "Strumenti" e poi su "Opzioni Internet…", vi apparirà una finestra come quella di figura 2. Scorrete l'elenco delle opzioni ed assicuratevi che le opzioni contrassegnate dal cerchietto rosso siano abilitate, qualora non lo fossero, abilitatele cliccandoci sopra. Confermate la scelta premendo il pulsante "OK". A questo punto potrebbe essere necessario riavviare il computer un'ultima volta.
Fatto questo il vostro computer è pronto a leggere il CD!

ESPLORARE IL CD

Inserite il cd nel vostro lettore e vedrete che sarà eseguito automaticamente Internet Explorer (o il browser che utilizzate normalmente), vi troverete quindi nella situazione di figura 3. A questo punto potete cliccare su una delle tre categorie principali: "Product Lines", "Application Notes", "Development Tools".
Fatto questo sarà caricato l'elenco delle categorie (proprio sotto l'elenco delle principali) e l'elenco dei documenti nello spazio bianco a sinistra, come in figura 4.
A questo punto il gioco è fatto, avete capito come navigare il cd e come accedere alla grande mole di documenti disponibili.

CONCLUSIONI

Come accennato precedentemente il Microchip's Technical Library, contiene tutta la documentazione relativa alle seguenti categorie di componenti:

- Componenti Analogici
- Microprocessori dsPIC®
- Memorie
- PICmicro®
- Dispositivi per radiofrequenza
- Dispositivi per la sicurezza
- Sistemi di sviluppo

Oltre alla documentazione (data-sheet, brochures e product brief) sono disponibili tutti i programmi necessari per programmare i PICmicro® (MPLAB, MPASM), gli aggiornamenti software per i sistemi di sviluppo e, forse la parte più interessante dell'intero CD, le Application Notes.
Le Application Notes, per chi non lo sapesse, sono dei documenti che spiegano come utilizzare un componente nei suoi diversi campi applicativi, spesso sono corredate di schema elettrico (alcune volte anche il circuito stampato) e di codice sorgente (nel caso si tratti di componenti programmabili).
Altre Application Notes, sono focalizzate alla risoluzione di algoritmi software, come per esempio: le routines matematiche, il controllo dei diversi tipi di bus, l'utilizzo delle periferiche interne e molto altro ancora.
Chiunque si cimenti con la programmazione dei PICmicro® troverà in questo CD una fonte inesauribile di informazioni mirate alla risoluzione di quasi tutti gli aspetti progettuali.

Figura 3

Un CD, questo, che arricchirà la vostra libreria e vi risparmierà le ore spese alla ricerca di informazioni e documentazione, soprattutto per quelli di voi che non hanno una connessione ADSL, ormai necessaria per scaricare documenti e software di dimensioni sempre maggiori.

Figura 4

mailbox

Dubbi, perplessità, malfunzionamenti, opinioni, commenti o richieste?

Inviateli a: **mailbox@farelettronica.com**

Oppure scriveta a:
Mailbox - Redazione di Fare Elettronica
Inware srl
Via Cadorna, 27/31 - 20032 Cormano (MI)

Le lettere più interessanti saranno pubblicate in queste pagine. Per quanto possibile, inoltre, cercheremo di dare una risposta privata a chiunque ci scriverà via email.

D PROBLEMI DI RICEZIONE

Sono un collezionista di radio a transistor e mi rivolgo a voi perché uno degli apparecchi della mia collezione, pur funzionando, non dà un rendimento soddisfacente ed inoltre emette dei forti sibili sia in corrispondenza di certe stazioni che in altri punti della scala dove non ci sono emittenti.
Finora le riparazione sui miei apparecchi le ho eseguite personalmente anche se le mie cognizioni tecniche sono in materia molto modeste.
In questo caso, nonostante vari tentativi non sono riuscito a trovare il guasto, per questo vi chiedo se potete indicarmi voi quale potrebbe essere in motivo di questo malfunzionamento.

Salavatore Cecere

R Il cattivo funzionamento del suo apparecchio è con ogni probabilità dovuto all'invecchiamento dei condensatori elettrolitici di disaccoppiamento e di fuga.
Le consigliamo di sostituire tutti i condensatori di questo tipo con altri nuovi.

D POTENZIOMETRO INSOSTITUIBILE

Sto riparando un ricevitore a valvole della fine degli anni '30 sul quale ho trovato il potenziometro del volume interrotto. Questo potenziometro ha un perno molto particolare e questo rende difficoltosa la sostituzione la sua sostituzione perché mi costringerebbe a fare un lungo e laborioso lavoro di adattamento.
Vi chiedo se avete qualche intervento da consigliarmi che mi consenta il ripristino del potenziometro guasto senza sostituirlo con altro nuovo.

Fabio Basento

R Dalle foto che lei ci ha inviato il potenziometro sembra essere del tipo con cursore a contatto indiretto, in questo caso e se la basetta non è distrutta, può provare a ricostruire la pista interrotta con della grafite. Per questo lavoro va benissimo l'anima di una comune matita che va sfregata sul punto dove la pista è interrotta fino a ricostruirla completamente.

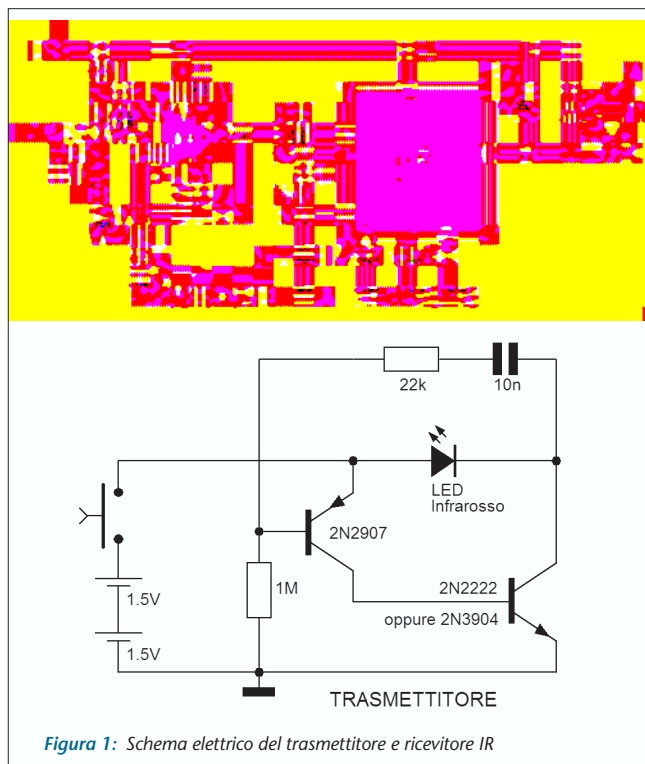
D COMANDO REMOTO IR

Avrei la necessità di realizzare un circuito che mi permetta di accendere/spegnere, tramite controllo remoto ir, le luci di ogni stanza delle casa in cui vivo.
Nella sala da pranzo sono presenti 2 interruttori separati, che controllano rispettivamente, 4 e 7 faretti alogeni, per illuminare parti separate della stanza, in cucina ho una plafoniera standard, come anche in camera da letto. Credo a questo punto che i circuiti da realizzare siano tre, uno per ogni ambiente. Essendo un appassionato alle prime armi, ma vostro lettore già da tempo, ho notato

con molto piacere che siete molto disponibili nell'aiutare, chi come me, non è in grado di raggiungere il proprio scopo con le minime nozioni a disposizione. Non chiedo di essere pubblicato, una risposta via mail andrebbe benissimo.
Nel ringraziarvi anticipatamente, vi saluto e vi faccio i miei più vivi complimenti per la pubblicazione della rivista più completa e ricca di argomenti che è in circolazione.

Simone Cavallari

R In figura 1 è riportata la coppia trasmettitore/ricevitore. Il trasmettitore invia un segnale infrarosso ad una determinata frequenza che viene riconosciuta dal ricevitore il quale, a sua volta, eccita un relé. Questo accorgimento evita l'eccitazione del relé da parte di altri trasmettitori o altri disturbi luminosi. Il trasmettitore non necessita di alcuna taratura, mentre il ricevitore deve essere tarato per riconoscere la frequenza del trasmettitore. La taratura si effettua puntando il trasmettitore ed agendo su R6 finché il relé non viene eccitato. È possibile che in condizioni di elevata luminosità, il ricevitore non sia in grado di ricevere il segnale del trasmettitore.

Figura 1: Schema elettrico del trasmettitore e ricevitore IR

D RILEVATORE DI CONSUMI

Vorrei realizzare un controllo su presa elettrica per osservare i consumi di elettrodomestici ad essa connessa, per questo ho ricercato trasduttori ad effetto hall del tipo: LEM LAH 25-NP come trasduttore di corrente ed LEM CV3-1000 come trasduttore di tensione.
Poiché questi componenti hanno un costo elevato, oltre ad essere difficilmente reperibili in commercio, vorrei chiedere se esistono soluzioni alternative.

Francesco Massara

R Il circuito di figura 2 permette di ottenere una tensione proporzionale alla corrente che scorre in un conduttore passante per il nucleo toroidale T1, quindi alla potenza assorbita dall'utilizzatore alimentato dallo stesso conduttore.
Il nucleo T1 è di tipo ANRA167 su cui sono avvolte 41 spire di rame smaltato da 0,8 mm e consente il monitoraggio di un conduttore del diametro esterno fino ad 1 cm.
Agendo sul trimmer P1 è possibile effettuare una taratura del circuito impiegando un carico di potenza nota (ad esempio una o più lampade da 100 W). La tensione di uscita può essere inviata all'ingresso analogico di un micro per effettuarne la conversione in digitale, quindi l'elaborazione del dato.
Sensore di corrente T1 può essere montato fino ad un massimo di due metri di distanza dal circuito. Collegando il sensore a valle del contatore ENEL e a monte di tutti i carichi, è possibile avere una stima del consumo globale (figura 3).

Figura 2: Schema elettrico del rilevatore di corrente

Figura 3: Utilizzo del sensore

NEWS 419

*Questo spazio è gentilmente offerto da **EONews**, il Quindicinale di notizie e commenti per l'industria elettronica di **VNU Business Publications Italia**.*

IDEE DI PROGETTO: LA NUOVA INIZIATIVA SUL WEB DI ELETTRONICA OGGI

"Idee di progetto – Design Ideas" è la nuova rubrica di Elettronica Oggi che è stata lanciata dal mese di Settembre 2003. Caratteristica saliente di questa nuova iniziativa è che sarà completamente ed esclusivamente on line ed accessibile dal sito www.ilb2b.it.
L'obbiettivo principale è creare una vera e propria libreria di idee alla quale tutti coloro che operano in maniera professionale nel mondo dell'elettronica possano "catturare" informazioni e suggerimenti utili per il loro lavoro quotidiano. Questo nuovo strumento vi permette di scaricare, con un solo click, tutte le risorse necessarie per risolvere velocemente qualsiasi problema e, in ultima analisi, minimizzare il time to market.

PROCESSORI AUDIO TV

Micronas ha reso noto che i processori audio Tv a chip singolo di ultima generazione della famiglia Msp 46xyK sono ora disponibili con uscite audio Pwm. Queste uscite sono state integrate soprattutto per connettere la famiglia in questione ad amplificatori audio di classe D switched mode. Azionati da segnali "Pwm ad alta risoluzione" Micronas, questi amplificatori offrono uno spettro di vantaggi rispetto ai dispositivi di classe Ab lineari tradizionali. Con un'efficienza fino al 95%, riducono in modo significativo la dissipazione termica ed eliminano la necessità di dissipatori di calore. Sprecandosi meno energia negli amplificatori audio, è possibile progettare apparati televisivi con minori alimentazioni, traducendosi in design più leggeri ed economici.

***readerservice.it** 419 - 51*

RELÈ A FIBRE OTTICHE

Di soli 31 (l) x 16 (largh.) x 9 (h) mm, il relè a fibre ottiche a basso profilo modello Wa di Matsushita presenta prestazioni interessanti, con perdite di isolamento contenute a 60 dB min. e perdite di inserzione massime di 1 dB (o .5 dB valore tipico). Disponibile in 2 configurazioni 1 x 2 e 2 x 2, è dotato di un prisma ottico azionato da un sistema magnetico polarizzato appositamente dimensionato; fornibile nelle versioni bistabile a 1 o 2 bobine, con assorbimenti contenuti a 150 mW, è in grado di lavorare in prima, seconda, terza finestra, fino a raggiungere lunghezze d'onda di 1550 nm. Due sono le connettorizzazioni disponibili per la versione standard (Sc/Adpc - Mu/Adpc), sono comunque fornibili su richiesta anche altri tipi di connettori a seconda delle specifiche esigenze.

***readerservice.it** 419 - 56*

FAMIGLIA DI SWITCHING

Zarlink Semiconductor (Unique Memec) ha lanciato una famiglia di nove interruttori Tdm/Tsi a bassa-media densità con la più vasta gamma industriale di caratteristiche integrate e programmabili, offrendo vantaggi in termini di prestazioni e diminuendo i costi in apparecchiature di reti wireless e cablate. Si tratta della serie Zl50021, che comprende tre interruttori a canale 1.024, tre interruttori a canale 2.048 e tre interruttori a canale 4.096 con fino a 32 ingressi e fino a 32 uscite o 32 flussi bidirezionali per interfacciarsi con componenti periferici a velocità di trasmissione dati da 2 a 16 Mbps. Con una gamma di opzioni prestazionali, supporta i progetti ad efficacia dei costi di apparecchiature in bassa larghezza di banda, quali server e concentratori ad accesso remoto, Ip-Pbx, small media gateway, stazioni base wireless e portanti di circuiti ad anello digitali.

***readerservice.it** 419 - 58*

REGOLATORI DI TENSIONE LINEARI

International Rectifier ha introdotto una serie di regolatori di tensione lineari Uldo a bassissima caduta di tensione e ad alta corrente, montati in contenitori a tenuta stagna e dedicati alle applicazioni aerospaziali e militari.
Si caratterizzano per una caduta di tensione compresa tipicamente tra 0,4 e 0,66 V. Presentano bassi ripple e rumore ed offrono una tolleranza di regolazione dell'1%. Sono regolabili da +20 a +1,8 V ed hanno una corrente d'uscita che spazia tra 3 e 10 A. L'intervallo delle tensioni d'ingresso è compreso tra +2,7 e +20 V.

***readerservice.it** 419 - 59*

MEMORIA FLASH SUPERAND DA512-MBIT

Renesas Technology ha annunciato HN29V512A, la serie di memorie flash superAND da 512-Mbit, che rappresenta la terza fase della gamma di prodotti superAND, successiva ai modelli da 128-Mbit e da 256-Mbit. La nuova serie offre un'elevata velocità di scrittura, di circa 4Mbyte al secondo, e maggiore semplicità di utilizzo, grazie alle funzioni per la gestione della memoria integrata.
La serie è disponibile in un package CSP dalle dimensioni ridotte (10 mm x 11,5 mm x 1,2 mm), come l'attuale modello da 256-Mbit. Ideale per essere utilizzata come memoria integrata per l'immagazzinamento di dati in prodotti consumer, come telefoni cellulari di prossima generazione, PDA, PC portatili e fotocamere digitali. La nuova serie utilizza un processo a 0,13 µm e la tecnologia AG-AND di Renesas Technology, che offre la possibilità di avere celle multi-level ad elevata velocità, approssimativamente pari a 4 Mbyte al secondo.

readerservice.it 419 - 60

SERIE DI DIODI ZENER

La nuova serie Bzx584Cxx-02 di diodi Zener resa disponibile Vishay Semiconductors (Silverstar-Celdis), classificata per un intervallo di tensioni da 2,4 a 15 V, è stata specificatamente progettata per applicazioni di regolazione della tensione vincolate da problemi di spazio in dispositivi di elaborazione e comunicazione portatili, comprendendovi telefoni cordless e cellulari, prodotti Bluetooth, videocamere digitali, lettori Mp3, computer notebook e Pda. Questi diodi presentano un valore di dissipazione di potenza di 200 mW, con resistenza d'isolamento di appena 600 K/W. Ognuno pesa solo 1,5 mg ed è disponibile in package plastici Sod-523, misuranti 1,2 x 0,8 mm con profilo d'altezza 0,6 mm.

readerservice.it 419 - 62

RICETRASMETTITORI DATI PROFIBUS RS-485/422

Intersil Corporation presenta due nuovi ricetrasmettitori di dati seriali ultraveloci RS-485/ 422 LINEARLINK™, ISL4486 e ISL81486.
Entrambi i circuiti integrati presentano configurazioni half-duplex che operano da un'alimentazione nominale di 5V su un ampio campo di tensione di tolleranza a +/-10% da +4,5 a 5,5 V, che riduce i requisiti di alimentazione e i costi. ISL4486 offre una velocità di trasferimento dati ultraveloce di 40 Mbps e un segnale differenziale di 2,5-V in un carico da 54 ohm. ISL81486 vanta una velocità di 30 Mbps ed una tensione di uscita differenziale di 2,3V.
ISL4486 e ISL81486 sono ricetrasmettitori con carico di 0,6 che consentono di predisporre 50 nodi su un bus, rispetto al limite di soli 32 nodi del normale carico a unità singola.

readerservice.it 419 - 63

MICROCONTROLLORE CON FLASH DA 16-BIT

Renesas Technology ha annunciato la disponibilità immediata del microcontrollore M16C/28 a 16-bit, il primo dispositivo a costo contenuto e con un ridotto numero di pin, che offre un ricco set di funzioni periferiche. Il chip da 64-pin (80-pin opzionali) include fino a 96 kbyte di memoria Flash, 4 kbyte di memoria virtuale EEPROM ed 8 kbyte di memoria RAM, ed offre svariate periferiche che precedentemente non erano disponibili in microcontrollori così piccoli (10 x 10 mm) e dal costo tanto contenuto. Il nuovo dispositivo è ideale per molte applicazioni consumer ed industriali, come elettrodomestici, controllo di motori, HVAC, sistemi di sicurezza e dispositivi medico-sanitari. M16C/28 offre svariate caratteristiche inusuali per un dispositivo della sua classe, che includono fino a cinque interfacce seriali in grado di offrire comunicazione sincrona, asincrona, I2C multi-master e modalità smart-card ISO7816; un ADC veloce (3.3 s) da 10-bit con un massimo di 24 canali.

readerservice.it 419 - 67

MEMORIE FLASH DA 32 A 256MB

Farnell InOne propone uno dei più ampi assortimenti di Semiconduttori, con una gamma completa di Memorie composta da oltre 1000 prodotti, suddivisi in 8 categorie: NVRAM, DRAM, SRAM, PROM, EPROM, EEPROM, FLASH MEMORY e PCMCIA MEMORY. Fra le novità da AMD la serie di Flash Memory MirrorBit™ con capacità da 32 a 256Mb. Questi prodotti appartengono alla famiglia Am29LV con tensione di alimentazione 3V e in package TSOP e BGA. Da ATMEL la famiglia AT45D Dataflash Memory con capacità da 1 a 64Mbits, su un range di tensioni operative compreso tra 2,7 e 3,6V, con package TSSOP, SOIC e TSOP.

Disponibile a catalogo da AMD e ST un'ampia varietà di Flash Memory delle famiglie 29F a 5V, 29W a 3V, con package SOIC e TSOP, capacità da 1 a 32 Mbits. Innumerevoli le EEPROMs offerte: si spazia da quelle con capacità a partire da 128-Bit fino ad arrivare a 256K, tensioni di alimentazione comprese tra 1,8 e 5V. I prodotti appartengono alle famiglie di tipo Seriale 24C, 24LC, 93C, 93LC, 524C, fino ad arrivare alle EEPROM Parallele della famiglia 28C.

readerservice.it 419 - 68

SISTEMI ON CHIP

Sharp Microelectronics ha presentato i nuovissimi modelli a 16/32 e 32 bit (Lh79524, Lh9525) della linea di prodotti BlueStreak con microcontrollori con tecnologia Arm e sistema on chip. Facilitano la connessione con la rete nonché l'operabilità tra i sistemi. Ciò è possibile grazie all'integrazione di Mac Ethernet e della funzione Usb su chip. Fanno parte delle soluzioni di sistema Sharp e si possono combinare facilmente con altri componenti realizzati da Sharp come Lcd, apparecchiature optoelettroniche e memoria flash.

readerservice.it 419 - 69

RIFERIMENTO DI TENSIONE

Il dispositivo Xicor

X60250 (Unique Memec) è un riferimento di tensione completamente integrato regolabile digitalmente. Unisce un riferimento di tensione a bandgap compensato in temperatura e un potenziometro digitale dotato di memoria non volatile, che insieme permettono di realizzare un riferimento di precisione regolabile nell'intervallo compreso tra 0,0 e 1,25 V.
L'integrazione di un riferimento di tensione di precisione ad 8 bit e di una funzione per la taratura diretta attribuiscono al dispositivo in questione diversi vantaggi rispetto alle alternative discrete adottate nelle applicazioni di produzione, taratura, messa a punto o regolazione.

readerservice.it 419 - 70

OPTOACCOPPIATORI AD ALTA VELOCITÀ

I due nuovi optoaccoppiatori ad alta velocità (Ps9814-1, Ps9814-2) offerti da Nec Electronics concernono dispositivi a canale singolo e doppio, in grado di operare a velocità di trasferimento di 10 Mbps, forniti in package So8. Possono essere utilizzati per convertitori Cc/Cc, apparecchiature di misura, controllo motori, sistemi di bus di comunicazione e numerose altre applicazioni e sono adatti per ambienti industriali gravosi. Entrambi hanno un Cmrr minimo di 10 k//micros, con valori tipici di 20 kV/micros; la distorsione dell'ampiezza di impulso max è meno di 50 ns, con valori tipici di solo 3 ns circa. Richiedono una tensione di alimentazione di 5 Vcc (7 Vcc max) sul lato rilevatore uscite per funzionare. La tensione d'isolamento è di 2500 Vrms.

readerservice.it 419 - 82

CONVERTITORE A/D

Adc08100 di National Semiconductor è un convertitore A/D ad elevate prestazioni che digitalizza i segnali con una risoluzione a 8 bit a una velocità di campionamento di 1,6 GHz e dissipa solamente 1,4 W con una tensione di alimentazione nominale a 1,9 V. Tali caratteristiche ne fanno il componente ideale in applicazioni quali strumentazione di misura e collaudo e apparati di comunicazione.

readerservice.it 419 - 84

FOTOSENSORE D'ILLUMINAZIONE AMBIENTE

Agilent Technologies ha presentato un fotosensore di illuminazione ambiente che prolunga l'autonomia delle batterie dei telefoni cellulari evitando di illuminare lo sfondo quando non è necessario e dimezza i costi e l'ingombro rispetto ai dispositivi precedenti. Oltre che per i telefoni cellulari, è ideale come soluzione di risparmio energetico per Pda, Pc palmari, lettori Dvd portatili, lettori Mp3, videocamere e fotocamere digitali. Siglato Hsdl-9001, consente un controllo preciso della soglia di rilevamento della luce. Le prestazioni sono garantite ad una tensione di alimentazione compresa tra 2,7 e 3,6 V e ad una temperatura da -25 a +85 gradi C.

readerservice.it 419 - 85

COME OTTENERE MAGGIORI INFORMAZIONI

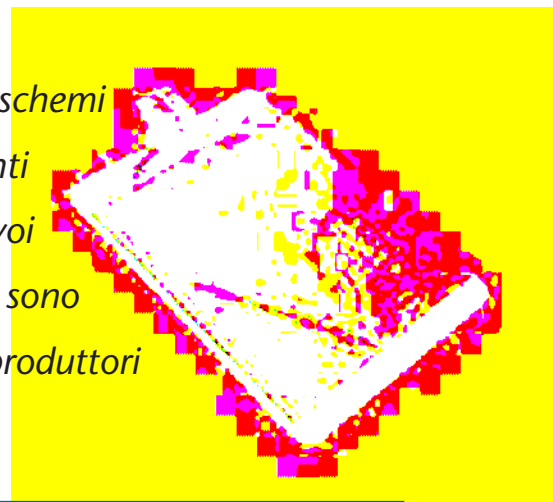
EONews offre il servizio "reader service" che vi consente, utilizzando l'apposito codice riportato alla fine di ogni news, di ricevere maggiori informazioni.

Visitate il sito **www.readerservice.it** e compilate la cartolina virtuale con i vostri dati, il numero della rivista, questo mese il **419**, ed i numeri di reader service presi dalle notizie che vi interessa approfondire.

EONEWS provvederà, tempestivamente, a contattare le aziende interessate, che invieranno al vostro indirizzo tutta la documentazione disponibile.

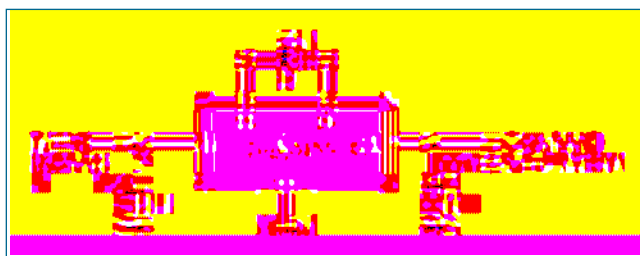
notepad

Dal blocco note di Fare Elettronica una raccolta di idee da tenere sempre a portata di mano

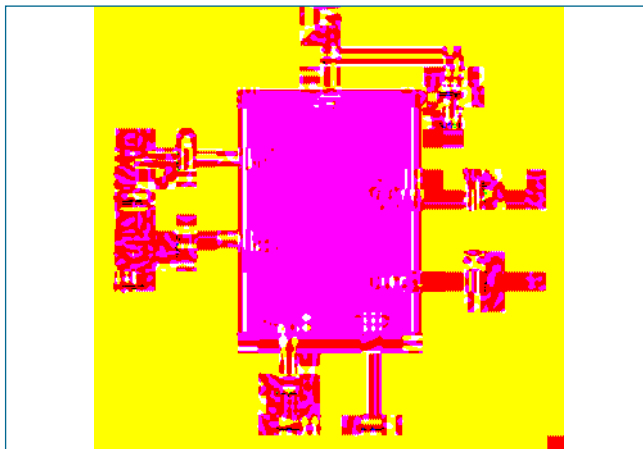
Questa rubrica ha lo scopo di fornire degli schemi applicativi o idee di progetto dei componenti elettronici più interessanti, selezionati per voi dalla redazione. Tutti gli schemi presentati sono elaborazioni di quelli ufficiali proposti dai produttori nella documentazione ufficiale.

TPS3600: BATTERY BACKUP 15

La famiglia TPS36xx comprende un serie di circuiti per il monitoraggio ed il controllo dell'attività di microprocessori in applicazioni low-power. In caso di problemi sulla linea di alimentazione il circuito provvede ad effettuare uno switch dell'alimentazione su una batteria di backup.

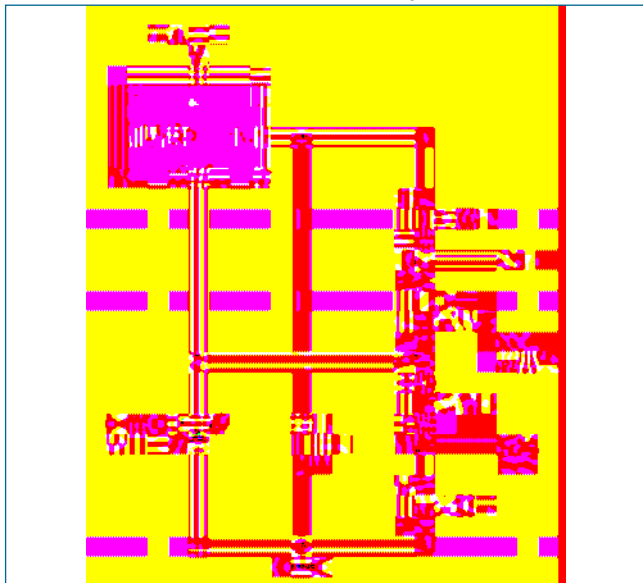
TPS60400: INVERTER A POMPA DI CARICA IN SOT-23 16

Il TPS60400 è un inverter a pompa di carica adatto ad applicazioni in cui è richiesta una tensione di polarizzazione negativa. Il componente è in grado di generare una tensione negativa non regolata, a partire da una tensione positiva di 1,6V (5,5V Max). La massima corrente di uscita è di 60mA.

TPA6101A2: AMPLIFICATORE ULTRA-LOW-VOLTAGE 17

Texas Instruments produce il TPA6101A2, un amplificatore stereo in grado di erogare 50mW di potenza su un carico da 16 Ohm, con una ten-

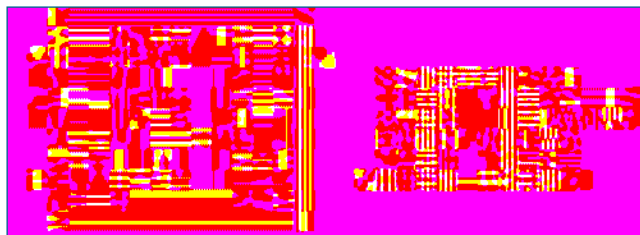
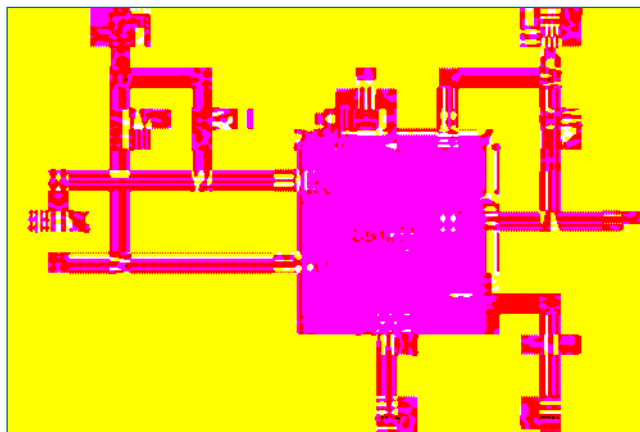
sione di alimentazione da 1,6V a 3,6V. Per il funzionamento sono richiesti solamente sei condensatori esterni. Il modello TPA6100A2 permette di impostare il livello di guadagno mediante resistori interni. Applicazioni tipiche di questi dispositivi sono in computer palmari, telefoni, ecc…

LM35: TERMOMETRO FAHRENHEIT

18

LM35 è un noto sensore di temperatura in grado di dare in uscita una tensione di 10mV/°C con una accuratezza di 0.5°C. Opera tra -55°C e

+150°C ed assorbe una corrente inferiore ai 60mA. In figura sono riportati i diversi package disponibili ed una applicazione come termometro Fahrenheit. In questa applicazione l'uscita è di tipo differenziale (non riferita a massa) e varia di 1mV per ogni grado Fahrenheit.

DS1307: REAL-TIME CLOCK I²C

19

È di Dallas Semiconductors questo clock real-time per bus I2C. Necessita solo del quarzo e delle eventuali resistenze di pull-up che devono essere dimensionate in base alla capacità parassita del bus. Questo chip comprende anche un calendario completo di giorno, mese ed anno. I giorni del mese e gli anni bisestili vengono gestiti automaticamente. Oltre alla possibilità di visualizzare l'orario nei formati 12/24h, il DS1307 effettua anche un monitoraggio della tensione di alimentazione commutando automaticamente su una batteria di backup in caso di interruzione. In figura, lo schema a blocchi del DS1307, il pinout e una tipica applicazione.

APPLICAZIONI EMBEDDED seconda parte

L'APPLICAZIONE INDUSTRIALE

Gianroberto Negri
info@gnrs.it

Prima di esaminare la struttura di un programma vedremo come creare una comunicazione mediante il protocollo UDP che, ci permetterà di testare quanto andremo a scrivere senza dover necessariamente avere una scheda collegata alla RS232 del PC.

PREMESSA: TECNICHE PER IL TEST DELL'APPLICAZIONE

L'applicazione industriale che ci accingiamo a realizzare per poter essere testata ha bisogno di una scheda collegata alla seriale del PC. Realizzeremo la stessa che, ci permetterà di dimmerare (far variare progressivamente) le luci di casa, più in là, tra qualche puntata. Occorre quindi qualcosa che possa sostituire tale scheda, almeno per il momento. Una soluzione potrebbe essere quella di collegare via seriale due PC ma. non tutti li posseggono. Esiste un modo per imbrogliare il programma di gestione e poter effettuare con tranquillità e sicurezza i nostri test. Si avvale di una tecnica che normalmente tutti utilizziamo quando ci colleghiamo con Internet ma, senza obbligarci ad impostare l'intera procedura di collegamento con tutti i parametri previsti.
Internet usa il protocollo **TCP/IP** per permettere la comunicazione tra i vari PC presenti nella rete. Un sottoinsieme dello stesso, denominato **UDP**, permette di avere gli stessi vantaggi ma senza complicarci troppo la vita. Per poter identificare un PC in maniera univoca nella rete, alla stessa stregua di un reale indirizzo di posta ci si avvale di un indirizzo IP composto da quattro serie di numeri separati da un punto. Sono indirizzi IP i numeri **255.255.255.0**, **197.0.1.0**, **192.188.182.1**. Esiste un IP che definirei magico che permette di non spostarsi dal PC che stiamo utilizzando ma che, dal punto di vista del comportamento e del modo di gestire il protocollo **UDP** è identico a quelli che identificano un PC della nostra rete locale o in Internet. Il numero è **127.0.0.1** ed è davvero comodo. Se due programmi che stanno girando contemporaneamente sullo stesso PC lo utilizzano possono parlarsi, scambiandosi messaggi, dati, comandi. È proprio quello che ci serve!

Come primo passo nella realizzazione della nostra applicazione vedremo come implementare un protocollo **UDP** in Visual Basic. Lo stesso verrà sostituito in seguito da quello che permetterà il colloquio con la scheda per la Domotica (dimmer delle luci). La cosa interessante è che la struttura del programma rimarrà pressoché la stessa, solo la parte di colloquio verrà sostituita.

Bene bando agli indugi e vediamo il programma che simulerà la scheda dei dimmer e già che ci siamo gli aggiungeremo anche alcuni CheckBox per simulare gli ingressi ed alcune immagini per simulare le uscite. Gli ingressi al loro variare provocheranno un cambio di stati al programma di gestione collegato a questo via **UDP**, mentre le uscite verranno attivate dai comandi inviati sem-

pre via **UDP** dallo stesso programma di gestione. Inizialmente realizzeremo un semplice scambio di valori, senza un vero e proprio protocollo e senza buffer dei comandi e delle risposte. La frase può sembrare sibillina… Abbiamo detto che l'**UDP** è un protocollo e ora diciamo che non utilizzeremo un protocollo! Vediamo di spiegare meglio il fatto: in realtà il protocollo **UDP** continuiamo ad utilizzarlo; quello che inizialmente non utilizziamo è un protocollo di secondo livello che "gira" ad un livello più alto dell'**UDP**. Si tratta del protocollo che specifica come devono essere impostati i comandi e le risposte per far in modo che il programma di supervisione e controllo e le periferiche presenti sul campo possano capirsi. In pratica si tratta di una convenzione su come vanno scritti i pacchetti che transitano sul BUS o più precisamente su come devono essere impostati i comandi e le relative risposte. Nel nostro attuale caso il BUS è costituito dal protocollo **UDP**.

I sorgenti dell'applicazione inerente all'applicazione che spiega il funzionamento dell'**UDP** sono scaricabili direttamente dal sito di Fare Elettronica e sono realizzati per la versione 6.0 di Visual Basic.

In figura 1 possiamo vedere i due programmi all'opera mentre si stanno scambiando informazioni. Si può notare come i CheckBox OUT 1, OUT 3, OUT 5, OUT 7 del Form Master A siano spuntati ed i rispettivi LED 1, 3, 5, 7 del Form Slave B siano accesi. Mentre i CheckBox OUT 2, OUT 4, OUT 6 ed OUT 8 del Form Slave B hanno acceso i rispettivi LED del Form Master A.

Cliccando sui rispettivi CheckBox nei due Form è quindi possibile accedere e/o spegnere i LED presenti negli stessi. Come accade il tutto? Vediamo di spiegarlo facendo riferimento ai sorgenti.

Iniziamo da quello inerente al Form MasterA e precisamente dal suo sottoprogramma Form_Load():

IMPORTANTE: assicurarsi di assegnare a RemoteHost il nome del computer, esso può essere espresso come: NNN.NNN.NNN.NNN (ES. 255.255.255.1)

Nel nostro caso l'IP che viene utilizzato fà rimanere all'interno del PC senza proiettarsi verso l'esterno nella rete.

Figura 1: I due programmi all'opera

```
Private Sub Form_Load()
 'Il nome del controllo
 ' è udpPeerA
 With udpPeerA
  '--------------------
  .RemoteHost = "127.0.0.1"
  'Porta a cui ci si
  'connette
  .RemotePort = 1001
  'Associazione alla porta
  ' locale.
  .Bind 1002
 End With
End Sub
```

In esso sono presenti le impostazioni per l'ActiveX WinSocket a cui abbiamo assegnato il nome di *udpPeerA*. È importante che lo stesso, agendo nella finestra delle proprietà al parametro Protocol, sia impostato su **UPD** e non su **TCP/IP** altrimenti non funziona.

L'istruzione *With* ha lo scopo di evitare di ripetere udpPeerA per ognuno dei parametri che andremo ad assegnare, infatti gli stessi sono presentati solo con un punto iniziale.

Il parametro *.RemoteHost* permette di impostare l'IP a cui collegarsi e nel nostro caso "127.0.0.1" che, come abbiamo detto rappresenta il nostro PC.

Il parametro *.RemotePort = 1001* imposta la porta a cui connettersi, cioè il canale da utilizzare per comunicare con l'altro programma.

Il parametro *.Bind 1002* imposta il canale locale, cioè il proprio canale di comunicazione.
Infine *End With* chiude la *With*.

Il sottoprogramma Form_Load() dell'altro Form è quasi speculare per cui analizzeremo solo le differenze:

```
Private Sub Form_Load()
 'Il nome del controllo è
udpPeerB
 With udpPeerB
  '--------------------
  .RemoteHost = "127.0.0.1"
  'Porta a cui ci si
  'connette
  .RemotePort = 1002
  'Associazione alla
  'porta locale.
  .Bind 1001
 End With
End Sub
```

Cambia ovviamente il nome del controllo ActiveX che in questo caso è *udpPeerB*.
Come si può notare il parametro *.RemoteHost = "127.0.0.1"* rimane lo stesso perchè vogliamo rimanere sullo stesso PC e non essere proiettati nella rete.
Invece *.RemotePort = 1002* cambia, come cambia *.Bind 1001*.
Riguardo questi due fondamentali parametri vanno spese due parole. Come si può notare, il parametro .RemotePort ha lo stesso valore di *.Bind* dell'altro Form e la cosa e reciproca.
Cosa vuol dire questo fatto? Semplice, che il canale definito in .RemotePort di un Form è collegato con il canale definito in *.Bind* dell'altro. In caso contrario non potrebbe avvenire il colloquio.
Vengono riservati due canali: **1001** e **1002** per poter fare in modo che i due programmi si parlino. Nessun altro programma può utilizzare tali canali e solo quando i programmi terminano vengono rilasciati.
Come già detto cambiando 127.0.0.1 con un'altro *IP* il tutto funziona lo stesso anche se un programma si trova in Italia e l'altro in Australia. Facciamo un esempio e ipotizziamo che il nostro PC abbia un *IP* di **255.198.192.1** e che il PC a cui vogliamo collegarci abbia un *IP* di **192.158.62.1**, i programmi rispetto a come sono ora dovranno essere modificati solo nella parte *RemoteHost* modificando rispettivamente nel programma che gira sul nostro PC l'*IP 192.158.62.1* e nell'altro l'*IP 255.198.192.1*.

Vediamo ora la parte inerente ai comandi da inviare allo Slave (Form Master B):

```
'------------------------
' Master A
'------------------------
Private Sub Out1_Click()
  Dim Cmd As String
  If Out1.Value = 0 Then
    Cmd = "OFF1"
  Else
    Cmd = "ON1"
  End If
  udpPeerA.SendData Cmd
End Sub

'------------------------
' Slave B
'------------------------
Private Sub Out1_Click()
  Dim Cmd As String
  If Out1.Value = 0 Then
    Cmd = "OFF1"
  Else
    Cmd = "ON1"
  End If
  udpPeerB.SendData Cmd
End Sub
```

Del sorgente spiegheremo solo due sottoprogrammi perchè gli altri 6 sono pressoché identici cambiano solo i comandi che vengono spediti.
È molto semplice, quando si clicca con il mouse ad esempio nella CheckBox OUT 1, viene richiamato il sottoprogramma *Out1_Click()* e viene testato se sia spuntato oppure no. Nel caso che non lo sia viene depositato nella variabile *Cmd* il comando "OFF1" altrimenti il comando "ON1", poi lo stesso mediante *udpPeerA.SendData Cmd* viene trasmesso attraverso il canale che si impostato allo slave.
Tutto qui. Ogni CheckBox richiama il proprio sottoprogramma che di conseguenza invia il comando ad esso riferito allo Slave.

Anche lo Slave funziona nella stessa identica maniera da come si può vedere dalla parte di sorgente riportata.

```
Private Sub
udpPeerA_DataArrival(ByVal
bytesTotal As Long)
 Dim strData As String
 udpPeerA.GetData strData
 Select Case strData
  Case "ON1"
   In1.BackColor = &H80FFFF
  Case "OFF1"
```

```
   In1.BackColor = &H8080&
  Case "ON2"
   In2.BackColor = &H80FFFF
  Case "OFF2"
   In2.BackColor = &H8080&
  Case "ON3"
   In3.BackColor = &H80FFFF
  Case "OFF3"
   In3.BackColor = &H8080&
  Case "ON4"
   In4.BackColor = &H80FFFF
  Case "OFF4"
   In4.BackColor = &H8080&
  Case "ON5"
   In5.BackColor = &HFF&
  Case "OFF5"
   In5.BackColor = &H80&
  Case "ON6"
   In6.BackColor = &HFF&
  Case "OFF6"
   In6.BackColor = &H80&
  Case "ON7"
   In7.BackColor = &HFF&
  Case "OFF7"
   In7.BackColor = &H80&
  Case "ON8"
   In8.BackColor = &HFF&
  Case "OFF8"
   In8.BackColor = &H80&
  Case Else
 End Select
End Sub
```

I comandi, una volta spediti, vengono rispettivamente ricevuti ed interpretati dal sottoprogramma *udpPeerA_DataArrival* mediante una *Select Case striata*. In questo caso sono stati previsti otto comandi per accendere i LED ed altri otto per spegnerli. Tutto qui.
Questa tecnica è valida per realizzare colloqui tra applicazioni, impostando un opportuno protocollo che definisca come vanno definiti i dati, i comandi e le risposte. Può essere utilizzato anche per gestire schede dotate di interfaccia di rete atte ad acquisire e gestire gli I/O del sistema.

Anche questa puntata è terminata, a rileggerci su queste pagine nel prossimo numero dove parleremo di come è strutturata l'applicazione e di come realizzarla in Visual Basic.

VIDEO-CONTROLLER CON MOTION DETECTOR

di Angelo Brustia
radio-bar@libero.it

Una alternativa ai controlli antiintrusione classici è quella di monitorare un ambiente con una telecamera ed elaborare il segnale video in modo da rilevare variazioni causate dal passaggio di una persona o, in generale, da qualunque cambiamento di scena ambientale.
Se poi c'è anche la possibilità di videoregistrare l'evento abbiamo realizzato un sistema di videosorveglianza, in grado di controllare fino a otto zone contemporaneamente.

L'uso dei microcontrollori come alternativa alle vecchie reti fisse sequenziali o combinatorie si sta diffondendo sempre più in campo hobbistico grazie anche al costo contenuto e alla continua disponibilità di modelli sempre più evoluti.
L'integrazione di periferiche sofisticate (AD converter, USART, Flash memory), unita alla velocità di esecuzione e alla facilità di upgrade del firmware permettono dunque di realizzare circuiti con funzionalità molto evolute e con pochi componenti, soprattutto facilmente modificabili agendo solo sul programma che "gira" sul micro.
In questa ottica ho pensato di realizzare un antifurto un po' particolare, la cui attivazione non dipenda da sensori volumetrici o perimetrici, ma dalla variazione dell'immagine rilevata da una o più telecamere (fino ad un massimo di otto) quando cambia qualcosa nella scena ripresa.
Questa tecnica viene utilizzata in campo professionale già da tempo e consiste nel confrontare una immagine video con un'altra memorizzata precedentemente: se i due frames differiscono oltre un certo limite scatta l'allarme. La memorizzazione delle immagini richiede convertitori AD veloci, molta RAM e soprattutto dei criteri per minimizzare il numero di campioni considerando, nel confronto tra frames, solo alcune zone dell'immagine.
Un circuito di questo tipo è solitamente molto complesso e richiede l'uso di tecniche DSP (Digital Signal Processing) per ottimizzarne le prestazioni; la mia proposta si basa invece sulla misura della variazione di livello del segnale video quando cambia la scena ripresa.
Il segnale video consiste di alcuni segnali di ampiezza ben definita e presenti a intervalli di tempo costanti (sincronismo verticale, orizzontale, indicatori di quadro pari e dispari) e di un segnale di ampiezza variabile legato alla luminosità della scena ripresa (il discorso vale sia per il B/N che per il colore): per ogni riga di scansione video sarà dunque disponibile un segnale che inizia con un impulso di sincronismo (orizzontale) e prosegue con una ampiezza legata alla luminosità fino all'impulso di sincronismo della linea successiva.
Il valore medio di questo segnale sarà dunque composto da un termine fisso (la media dei segnali di sincronismo) e da un termine variabile (luminosità): analizzando quest'ultimo si potrà stabilire se la scena inqua-

drata ha subito variazioni, cioè se "qualcosa si è mosso".

SCHEMA A BLOCCHI

Il circuito consiste di otto interfacce per telecamera ed una "scheda madre" per il sistema di controllo e l'alimentazione.
Lo schema a blocchi della singola interfaccia (scheda telecamera) è rappresentato in figura 1 ed il corrispondente circuito elettrico è riportato in figura 2

Il segnale proveniente dalla telecamera viene amplificato (blocco 1) in modo da compensare eventuali attenuazioni esterne; il guadagno dello stadio è pari a 1+[(R3+V1)/R2] quindi varia da uno a sette volte in ampiezza a seconda della posizione del trimmer V1 (controllo della sensibilità).
Nel caso si vogliano rilevare variazioni minime della scena ripresa tale componente va regolato per avere il massimo guadagno, tuttavia, da prove fatte, una regolazione a metà corsa va bene in quasi tutte le situazioni di impiego.

Figura 1: *schema a blocchi scheda telecamera*

Si noti che non è necessario utilizzare operazionali video a larga banda per questo stadio perché, pur avendo il segnale video una banda piuttosto estesa (circa 5 MHz), la scena ripresa è statica e non occorre avere una assoluta fedeltà nella risposta, ma solo individuare variazioni relative di livello.
Per avere maggiore stabilità nella risposta è stato inserito il condensatore C2 che limita la banda passante dell'amplificatore e previene eventuali autooscillazioni.
Il segnale video, dopo il disaccoppiamento operato dallo stadio precedente, deve ora essere raddrizzato per fornire una componente continua di tensione utile per verificare variazioni di immagine.
Il raddrizzamento non può essere effettuato con un normale diodo per via della sua tensione

Figura 2: *schema elettrico scheda telecamera*

Elenco componenti - Scheda telecamera	
Sigla	**Valore**
C1, C7, C11, C12, C14, C15	1 µF 63 V poliestere
C2	2.7 pF ceramico
C3÷C6, C17, C18	0.1 µF SMD ceramico
C9	10 nF ceramico
C13	10 nF 63 V poliestere
C8	10 µF 16 V tantalio
C10	2.2 µF 63 V poliestere
C16	47 nF 63 V poliestere
C19, C20	47 µF 16 V elettrolitico
R1	75 Ω 1/4 W
R2, R3	1 KΩ 1/4 W
R9, R13	1 KΩ SMD
R4	2.2 KΩ 1/4 W
R5, R10	100 KΩ 1/4 W
R6, R14, R15, R18, R20	10 KΩ 1/4 W
R7	1.2 MΩ 1/4 W
R8	5.6 KΩ 1/4 W
R11	1 MΩ 1/4 W
R12	15 KΩ 1/4 W
R16, R17	220 Ω 1/4 W
R19, R21	4.7 KΩ 1/4 W
V1	Trimmer verticale 4.7 KΩ
V2	Trimmer verticale multigiri 4.7 KΩ a regolazione laterale
D1, D2	1N4148
P1÷P3	Connettore maschio per c.s. passo 2.54 mm
U1	TL081
U2	LM324

di soglia (circa 0.7 V) che introdurrebbe una distorsione di linearità inaccettabile sui bassi livelli di segnale; il circuito che meglio si presta a questo scopo è quello formato dall'operazionale U2b, da D1, R4 e dai condensatori di filtro C7-C9 (blocco 2) ed è noto come "raddrizzatore di precisione". Utilizzando questa configurazione avremo una risposta perfettamente lineare a partire da 0 V solo per la polarità positiva del segnale.

Il condensatore C8 dovrebbe essere ad alta stabilità (Tantalio), il condensatore C7 serve a ridurre l'induttanza equivalente di C8 (particolarmente elevata) in modo da avere una risposta esente da oscillazioni e C9 rappresenta un by-pass per le frequenze più alte; infine la resistenza R5 scarica parzialmente i condensatori in modo che la tensione livellata non rimanga costantemente al valore massimo, ma sia proporzionale al valor medio del segnale di ingresso.

La tensione presente ai capi della capacità di filtro è conseguenza dei segnali di sincronismo e del segnale video; purtroppo la componente media legata ai segnali di sincronismo è decisamente più elevata di quella determinata dalla variazione di luminosità, occorre quindi trattare il segnale video; per estrarre solo questa informazione.

Il condensatore C10 ha il compito di far passare solo le variazioni di tensione che conseguono ad un cambiamento della scena ripresa bloccando nel contempo la componente continua dovuta al raddrizzamento dei segnali di sincronismo che, essendo costanti nel tempo, danno origine ad un livello di tensione costante.

Il livello di queste variazioni è decisamente basso (qualche mV) e occorre una forte amplificazione per raggiungere un livello più elevato (circa 1.5 V) necessario per avere una buona dinamica per il successivo convertitore AD; oltre ad amplifica-

RIVOLUZIONE DELLA COMUNICAZIONE
2 - 3 OTTOBRE 2004
PARCO ESPOSIZIONI NOVEGRO

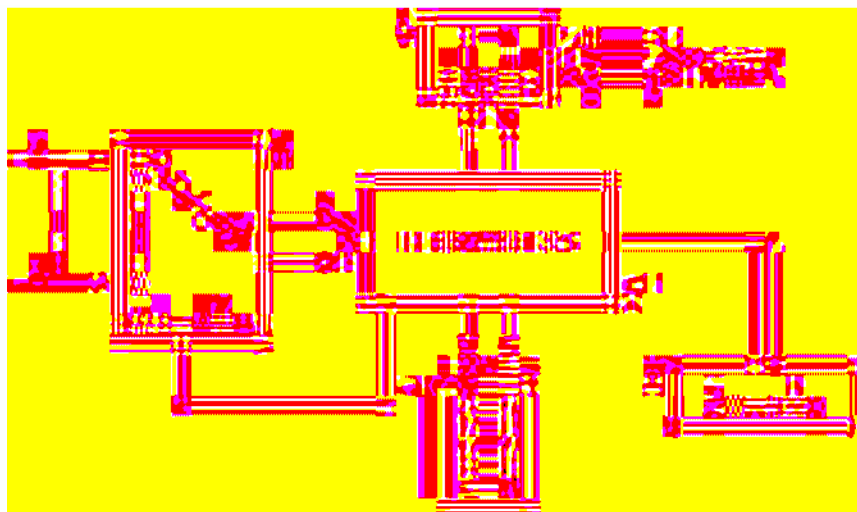



Figura 3: schema a blocchi Unità di controllo

re occorre inoltre eliminare tutti quei disturbi che potrebbero essere scambiati per variazioni dell'immagine, ad esempio quelli generati da fonti luminose alimentate a 50 Hz.

Il circuito che meglio si presta a soddisfare entrambe le esigenze è un filtro attivo passa-basso a retroazione multipla costruito attorno agli operazionali U2a e U2d (blocco3).

Dato che il guadagno necessario al funzionamento del circuito è di circa 62 dB è meglio realizzare due amplificatori separati con

Figura 4: schema elettrico Unità di controllo

più basso guadagno in modo da avere una maggiore stabilità di funzionamento, entrambi di tipo invertente per non avere rotazione di fase tra ingresso e uscita. Con i componenti indicati nello schema la banda passante a –3dB è di 2.7 Hz con una attenuazione di circa 48 dB alla frequenza di 50 Hz.

Elenco componenti - Unità di controllo	
Sigla	**Valore**
C1, C4÷C7, C15, C19÷C22	0.1 µF SMD ceramico
C16, C17	22 pF SMD ceramico
C2, C3	100 µF 16 V elettrolitico
C8, C9, C18	22 µF 16 V SMD elettrolitico
C10, C11, C12, C13, C14	4.7 µF 50 V elettrolitico
R1, R2	4.7 KΩ 1/4 W
R3	4.7 KΩ SMD
R4	1 KΩ 1/4W
R5, R6, R7, R8	10kΩ 1/4W
R9	68 Ω SMD
L1	Induttanza 68 µH
D1	Diodo Shottky 1N5817
D2, D3	1N4007
D4	1N4148
Q1	BC237B
Q2	7805
U1	MAX764
U2	PIC16F877A
U3	MAX4315
U4	MAX232
X1	Quarzo 4 MHz
P1_0, P1_7	Connettore femm. per c.s. passo 2.54 mm
P2_0, P2_7	Connettore femm. per c.s. passo 2.54 mm
P3_0, P3_7	Connettore femm. per c.s. passo 2.54 mm
P4, P5, P6, P7	Connettore maschio per c.s. passo 2.54 mm
Sw1	Pulsante miniature per c.s. normalmente aperto
Sw2	Deviatore da pannello 1 via 4 posizioni
Conn.1	Connettore D 9 poli femmina
RL1	Relé miniatura per c.s. 5V 2 scambi

Il segnale all'uscita del filtro può assumere sia valori positivi che negativi dal momento che il condensatore C10 esegue una operazione matematica di derivazione: se la tensione in uscita dal raddrizzatore cresce la tensione sarà positiva, viceversa se decresce (la polarità è quindi determinata dal modo di variazione dell'immagine).
Purtroppo il convertitore AD del PIC 16F877A accetta solo valori positivi di tensione quindi occorre sommare (blocco 4) una tensione (bias) che, per questioni di dinamica, deve essere almeno pari all'escursione di tensione massima negativa in modo da spostare il range di variazione da -Vmax÷Vmax all'intervallo positivo 0÷2Vmax.
Con questo accorgimento, assumendo un bias di 1.5 V, la minima tensione sarà ora di circa 0 V e la massima di circa 3 V; la regolazione dipende dal trimmer V2.
Il diodo D2, infine, evita che giunga una tensione negativa al convertitore AD nel caso il bias sia troppo basso.
Le schede telecamera potranno essere pari ai canali di conversione AD del micro (cioè otto se si impiega il 16F877A, ma potrebbero anche essere di più riprogettando il circuito con un altro micro della famiglia 18F sempre della Microchip).
Se si desidera invece modificare il circuito per utilizzare un numero inferiore di ingressi occorre inibire alcune subroutine nel sorgente assembler (come mostrato più avanti); l'operazione è molto semplice e consiste nell'aggiungere qualche commento (";")

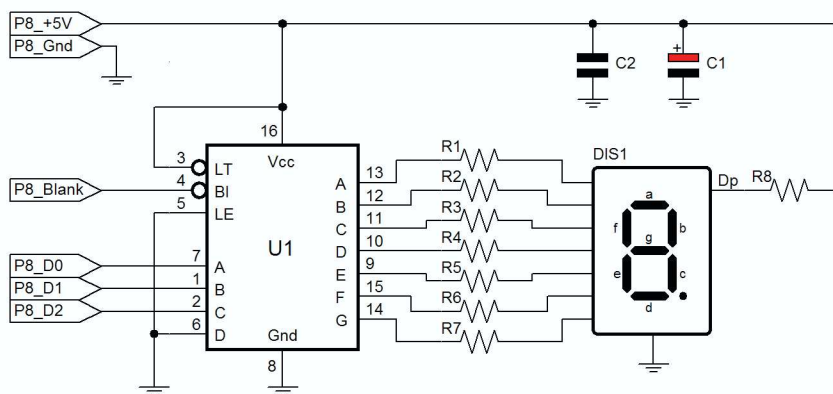



Figura 5: Schema elettrico scheda display

nel codice, quindi è fattibile anche da chi è a digiuno di linguaggio assembler; naturalmente se la variante può essere utile a più persone posso modificare il codice introducendo dei ponticelli esterni da aprire o chiudere in funzione del numero di telecamere collegate.
La figura 3 rappresenta lo schema a blocchi del "cuore" del circuito, ovvero l'unità di controllo

Elenco componenti Scheda display	
Sigla	**Valore**
C1	22 μF 20 V SMD elettrolitico
C2	0.1 μF SMD ceramico
R1÷R8	470 Ω SMD
U1	CD4511
Dis1	Display catodo comune FND500 (o equivalente)
P8	Connettore maschio per c.s. passo 2.54 mm

gestita dal microcontrollore e la figura 4 il corrispondente schema elettrico.

Ognuno dei segnali provenienti dalle telecamere viene inviato ad una scheda video e, contemporaneamente, al blocco 6 (multiplatore video) costituito dal circuito integrato U3 (Max 4315).
Questo chip prodotto dalla Maxim è un completo mixer video analogico a otto canali in grado di effettuare la commutazione di canale sulla base del numero binario presente sugli ingressi A0÷A2 gestiti dal microcontrollore; in presenza di una condizione di allarme il micro stesso provvede anche ad abilitare (tramite il pin SHDN) il passaggio del segnale video tra l'ingresso selezionato (IN0-IN7) e l'uscita OUT.
Il multiplexer, prodotto anche nelle versioni a 2 e 4 canali, è caratterizzato, oltre che da una notevole banda passante (circa un centinaio di MHz), dall'assenza dei disturbi dovuti al cambio di canale (glitch) e dalla possibilità di funzionare con alimenta-

Figura 6: Flow chart programma microcontrollore

zione singola o duale (nel nostro caso si usa la duale poiché il segnale video standard ha una tensione minima inferiore a 0 V).
Ho preferito questa soluzione a quella "classica" che fa uso degli switch cmos HC4066 come commutatore video poiché è molto più compatta e l'elevata banda passante permette una risoluzione spaziale migliore, quindi una qualità di visione superiore.
I diodi D2 e D3 che si vedono in serie all'alimentazione servono solo a ridurre di circa 1,4 V la tensione di alimentazione dato che alimentando il circuito direttamente con +5V e –5V si corre il rischio di lavorare troppo vicino alla massima tensione di funzionamento dichiarata (10.5 V); del resto il segnale ha una variazione limitata a circa 2 V quindi per garantire una adeguata dinamica basta che l'alimentazione non sia inferiore a 3 o 4 V.
Il segnale DC proveniente dalle schede di acquisizione video viene inviato al blocco 7 e precisamente ai pin AN0÷AN7 corrispondenti agli otto canali analogici del convertitore A/D presente nel micro (i canali fanno capo ad un multiplexer interno la cui selezione viene effettuata dal firmware).
In condizione di allarme viene attivato il relé RL1 (utilizzabile per comandare un videoregistratore) e viene visualizzato sul display (blocco 9) il numero del canale in allarme; il processore, tramite la porta seriale facente capo ai pin RC6, RC7 e al translatore U4 (MAX232) trasmette, tramite collegamento seriale, l'indicazione di una condizione anomala al PC remoto di controllo.
In figura 5 è riportato il circuito elettrico della scheda display.

DESCRIZIONE DEL SOFTWARE DI GESTIONE

Il codice assembler che "gira" sul micro è stato scritto con l'ottimo ambiente di sviluppo MPLAB messo a disposizione gratuitamente da Microchip.
Lo schema a blocchi del programma è riportato in figura 6.
Per brevità non è possibile riportare il dettaglio del listato, ma il programma è molto commentato e credo sia abbastanza comprensibile.
Vorrei sottolineare il fatto che il micro scambia una serie di dati con il PC di controllo del sistema con cui dialoga continuamente tramite seriale; l'elevata velocità (19,2 kb/s) permette di vedere le variazioni in tempo reale, a condizione che il PC sia abbastanza veloce (le prove sono state fatte con un Pentium III a 600 MHz e con un Pentium a 166 MHz e le differenze erano veramente notevoli!).
Nel caso la linea di comunicazione sia lunga più di 5m consiglio di abbassare la velocità a 9600 b/s anche se, per motivi di sicurezza, i dati vengono trasmessi più volte di seguito, quindi, anche senza controlli Xon/Xoff, l'affidabilità rimane comunque elevata.
Nel caso non interessi l'utilizzo di otto telecamere è importante modificare (come già detto in precedenza) il programma assembler per inibire il controllo delle porte inutilizzate che, per via della tensione troppo bassa, (quindi fuori dalla finestra di "normalità") sarebbero in condizione di allarme permanente.
Per fare ciò occorre:

- Aprire il file scansione41.asm (ad esempio con MPLAB).
- Abilitare l'indicazione dei numeri di riga.
- Andare alla riga 138 (subroutine "campiona canali").
- Inserire un ";" all'inizio delle linee relative alla conversione AD per i canali non utilizzati (ad esempio le linee 183-184-185 se non si vuole utilizzare il canale 5).

```
Movlw 0xFF
Movwf  ch5_on
Movlw  0x05
Movwf  canale
Call   conversioneAD
Btfss  STATUS,C
Clrf   ch5_on
```

- Ricompilare il tutto e programmare il micro.

L'applicativo di controllo è stato scritto in Visual Basic 6.0 ed è scaricabile dal sito di Fare Elettronica assieme ad un controllo OCX di pubblico dominio scritto da Pierluigi Pagnoni (che ringrazio per averlo reso disponibile) che simula un diodo LED; il file Ledbutton.ocx va inserito nella directory dei componenti Visual Basic (solitamente c:\windows\system o simili).

Protocollo di comunicazione PIC-PC

Ritornando al codice assembler riporto il protocollo di scambio dati (stringhe di interrogazione e risposta) in modo che ognuno possa scrivere un proprio applicativo (i numeri sono HEX quindi 786403, ad esempio, è da intendersi come tre numeri ad otto bit 78 60 03 in successione;

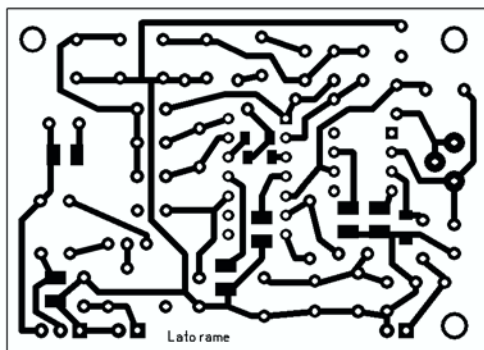



Figura 10: *Circuito stampato scheda telecamera in scala 1:1 (lato rame)*

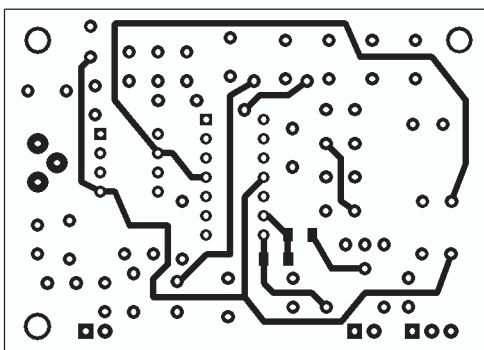



Figura 12: *circuito stampato scheda telecamera in scala 1:1 (lato componenti)*

Figura 11: *Circuito stampato scheda telecamera disposizione componenti*

il + tra i dati indica semplicemente che i dati sono in successione uno dopo l'altro).
Le varie sequenze "header" che identificano insiemi specifici di dati (es. 786403) sono scelte in modo da essere abbastanza ortogonali tra di loro e quindi minimizzare il rischio di lettura errata dei dati che seguono da parte del software.

Invio richiesta dei parametri di setting (residenti in EEPROM): all'invio da parte del PC della stringa "FF" il PIC risponde con la stringa 786403+delay1..4+epsilon+0+0+0 dove delay1..4 sono i

Figura 8: *Esempio di Interfaccia Utente con active-X commerciali*

Figura 7: *Interfaccia Utente programma di gestione*

Figura 9: *Oscilloscopio a 8 tracce realizzato con active-X commerciali*

Figura 13: *Circuito stampato unità di controllo in scala 1:1 (lato rame)*

tempi di attivazione del relé in caso di allarme e sono associati alle posizioni del deviatore Sw2; in altre parole se viene settato, ad esempio, delay1=10 e il commutatore è in questa posizione, in caso di allarme il relè rimane eccitato per 10 sec e quindi all'uscita Out è disponibile, sempre per 10 sec, il segnale video della telecamera che ha generato l'allarme. Il campo di validità va da 0 a 200 sec.

Epsilon indica metà ampiezza della finestra di tensione entro cui non scatta l'allarme (la finestra ha quindi ampiezza 2 x epsilon) ed è settabile da 0 a 1.96 V attorno al valore statico di 1.5V (tensione di uscita dei moduli telecamera in assenza di allarme), brevemente la finestra è centrata su 1.5V e può variare tra +/- 0 V e +/- 1.96 V quindi l'escursione massima consentita è di 3.92 V. Il campo di validità è tra 0 e 200; nell'applicativo VB viene mostrato, per comodità di lettura, direttamente il valore di tensione assumendo che a 255 corrisponda il valore di 5V.

Invio sequenza completa dal PIC: il micro invia continuamente al PC sequenze di aggiornamento lunghe 11 byte con la seguente struttura (gli 0 hanno il solo scopo rendere uguale la lunghezza delle varie stringhe):

786403+delay1+delay2+delay3+delay4+epsilon+0+0+0
23FD0A+valore letto A/D0+……..+valore letto A/D7
6E024A+numero canale in allarme+0+0+0+0+0+0+0

La prima sequenza è già stata descritta, la seconda invia i valori letti in tempo reale dai convertitori analogico/digitale di ogni canale (da 00 a FF corrispondenti all'intervallo 0-5V, anche se, per evitare problemi con le sequenze identificative, la dinamica è stata ridotta), la terza indica il numero del canale in allarme; nel caso ci fossero

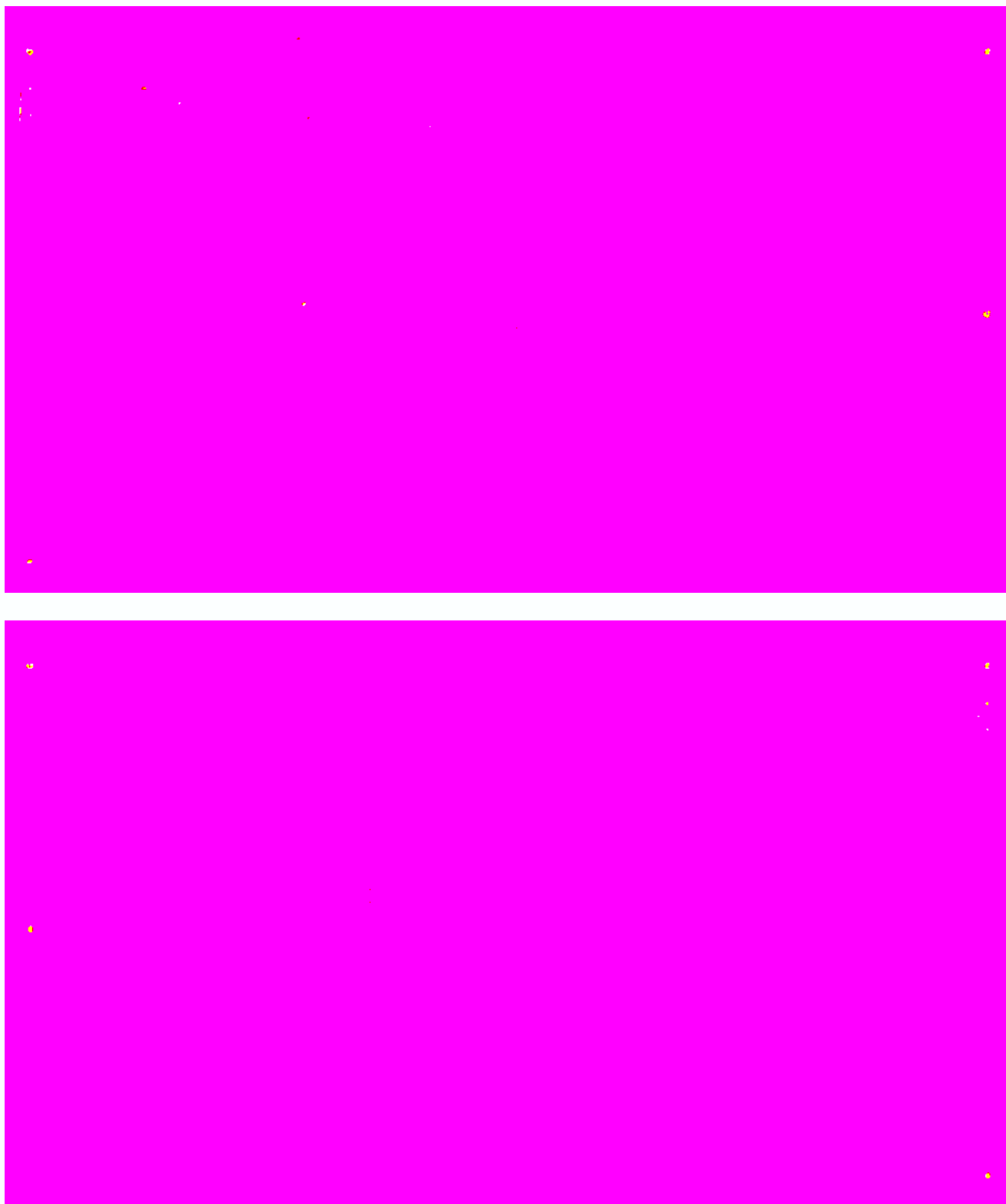



Figura 14: Circuito stampato unità di controllo disposizione componenti

più canali in allarme simultaneo questi vengono mostrati in successione solo se la condizione di allarme persiste dopo che il canale precedente è ritornato in posizione di riposo; nel caso persista invece la condizione di allarme, al termine del tempo impostato il canale rimarrà atti-

vo fino a che la scena ripresa non varia più.

Invio parametri di settaggio da PC a micro:

FC+delay1+delay2+delay3+delay4+epsilon

SOFTWARE DI CONTROLLO

Il programma in Visual Basic sviluppato per la gestione (figura 7) comprende una serie di slider per impostare i tempi di visualizzazione e l'ampiezza della finestra (nel caso si desideri caricare il default basta premere "Tempi standard").

Figura 15: *Circuito stampato unità di controllo in scala 1:1 (lato componenti)*

Dopo aver programmato i tempi e l'ampiezza della finestra si preme il pulsante "Aggiornamento"; l'accensione del LED verde indica che i parametri sono stati trasmessi e l'aggiornamento delle finestre "Delay1. Ampiezza metà finestra" è la conferma che sono stati memorizzati nella EEPROM del micro (i dati in queste finestre sono letti dallo stream inviato dal micro); nel caso il LED rimanga rosso significa che non ci sono dati di ritorno sulla seriale quindi il collegamento è mancante o si è interrotto (oppure non è stato premuto il pulsante di aggiornamento).

Il livello della soglia va fissato ponendola al valore massimo (3.92V) in modo che nessuno dei canali vada in allarme, quindi si abbassa gradualmente lo slider sino al verificarsi di un allarme (ovviamente le telecamere devono inquadrare una

Figura 16: *circuito stampato scheda display in scala 1:1 (lato rame)*

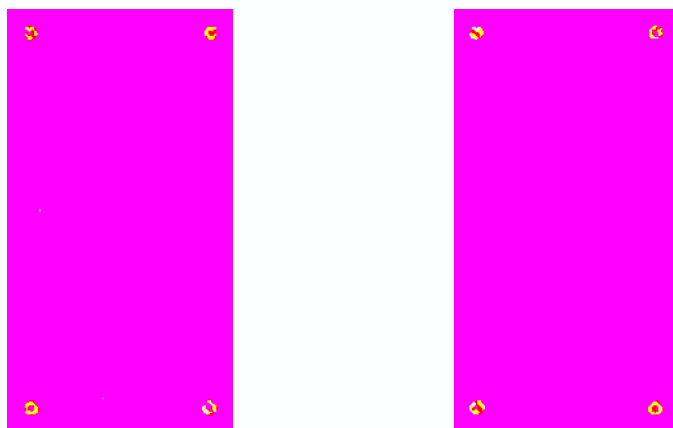



Figura 17: circuito stampato scheda display disposizione componenti

Figura 18: circuito stampato scheda display in scala 1:1 (lato componenti)

scena fissa e il superamento della soglia sarà causato dal rumore casuale di fondo); consiglio, in questa fase, di porre a 0 sec i tempi di visualizzazione per velocizzare la taratura.
Stabilita la soglia minima questa potrà essere alzata dipendentemente dal livello di sensibilità richiesto: più la soglia è elevata, maggiore dovrà essere il movimento nell'immagine rilevata dalla telecamera quindi minore la sensibilità.

Il programma, come si vede, è molto "essenziale" e può essere migliorato e ampliato ricorrendo a componenti Active-X commerciali per Visual Basic.
A titolo di esempio in figura 8 e figura 9 è riportato il form di un programma analogo al precedente (con in più un utilissimo oscilloscopio a otto tracce per vedere l'andamento del segnale) realizzato con componenti reperibili in Rete (purtroppo sono scaricabili solo in Demo e funzionano per 10 minuti!).

È stato previsto un controllo "doppio" dei tempi di allarme, software e da pannello, per potere utilizzare il circuito anche "stand-alone" senza il controllo da remoto; in questo caso si settano i tempi di intervento per ogni posizione del commutatore (con gli slider di cui sopra) e i dati, residenti in EEPROM, rimangono associati al controllo manuale anche dopo aver spento l'apparato o averlo scollegato il PC.

NOTE DI MONTAGGIO

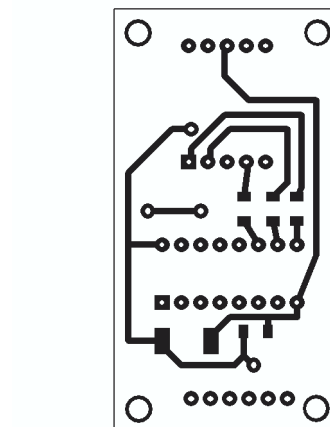
Le figure 10, 11, 12 riportano entrambe le facce del circuito stampato con la disposizione dei componenti della scheda telecamera, le figure 13, 14, 15 quelli dell'unità di controllo e le Figure 16, 17, 18 quelli della scheda display.
Per la preparazione dei circuiti stampati a doppia faccia consiglio caldamente di utilizzare la fotoincisione anche perché permette di incidere simultaneamente i circuiti stampando uno o due fogli master di acetato.
Il montaggio dei componenti non presenta grosse difficoltà; eventuali problemi possono nascere dalla saldatura degli SMD per i quali consiglio, magari utilizzando una lente, di stagnare una delle piazzole, appoggiare il componente, saldarlo tenendolo fermo con una punta, saldare il secondo reoforo sull'altra piazzola: attenzione a non esagerare con il calore perché potrebbe staccarsi la metallizzazione !
Particolare cura dovrà essere posta nel caso i componenti siano di recupero per evitare di rovinarli scaldandoli troppo (trovare in commercio gli SMD sciolti, non in reel da migliaia di pezzi, è piuttosto difficile e costoso; conviene quasi sempre recuperarli da vecchie schede).

Tutti i condensatori dovranno essere di ottima qualità (poiché determinano la stabilità termica del circuito), vanno bene ad esempio i tipi poliestere "a cubetto".
Per evitare inneschi dovuti all'elevato guadagno, le schede telecamera dovranno essere schermate inserendo tra di loro del lamierino di ottone (va bene anche un ritaglio di vetronite doppia ramatura); la scheda di controllo è già predisposta per

Figura 19: *Scheda telecamera lato componenti*

Figura 20: *Scheda telecamera lato saldature*

le schermature che dovranno essere saldate su entrambi i lati (questa operazione va eseguita dopo aver montato tutti i componenti altrimenti diventa impossibile raggiungere alcune piazzole con il saldatore).
La figura 19 mostra la scheda telecamera dopo l'assemblaggio, il retro del c.s. con i condensatori SMD è riportato in figura 20.
Il montaggio della scheda di controllo è un po' più complesso delle schede video perché occorre saldare diversi connettori in cui andranno inserite queste ultime e realizzare diverse connessioni tra le piazzole del lato componenti e del lato saldatura (figura 21).
Questa scelta è stata dettata dalla necessità di ottenere un circuito facilmente riproducibile: l'ideale sarebbe stato realizzare un c.s. a fori metallizzati, magari a tre strati, ma avrebbe determinato grosse difficoltà di realizzazione.

Queste operazioni devono essere eseguite prima di inserire gli altri componenti e le schermature, in particolare occorre allineare i pin di ogni scheda in modo che siano bene in asse (consiglio di infilare i connettori nei pin di una scheda, inserire il tutto nel c.s. e saldare, in tal modo i pin sono sicuramente allineati).
Il mixer video è in contenitore SMD a passo ridotto TSSOP16 quindi occhio a non creare un unico cortocircuito tra i pin; se possibile utilizzare un saldatore a spillo oppure modificare una vecchia punta saldandovi un chiodino sottile (a castolin ovviamente, non a stagno!); la figura 22 riporta il lato saldature del c.s. con il particolare dell'integrato (i fili ed il diodo incollato al circuito che si vedono nella foto sono poi stati eliminati nel circuito definitivo).

Figura 21: *Particolare del connettore*

Figura 22: *Particolare del mixer video*

Figura 23: *Particolare schermature e connettori per schede video*

Figura 24: *Unità di controllo lato componenti*

Il micro è meglio sia su zoccolo in modo da poterlo programmare agevolmente, in alternativa si può usare la programmazione ICSP sui pin RB6 e RB7 che sono stati lasciati liberi nell'ottica di sviluppi futuri del software.
La bobina L1 dell'alimentatore switching non è critica, va bene qualunque valore tra 40 e 80 microHenry, meglio se è realizzata su un piccolo toroide (con filo diametro 0.5 mm) così si evita che il flusso disperso influenzi le schede video ad alto guadagno.
I contatti del relé sono messi in parallelo in modo da poter commutare correnti elevate (se si usa il relé per accendere il videoregistratore assicurarsi che sopporti 220V in quanto quasi tutti i modelli di relé da c.s. sono garantiti per 125 V); per comandare il registratore è meglio utilizzare l'ingresso da comando esterno a bassa tensione disponibile su quasi tutti i modelli più recenti.
Il micropulsante Sw1, infine, serve a resettare il micro in presenza di condizioni anomale di funzionamento (non dovrebbe servire, ma con il software non si sa mai) o nel caso sia stato settato per errore un tempo di allarme eccessivamente elevato.
In figura 23 si vede l'Unità di controllo senza le schede video inserite.
La figura 24 e la figura 25 mostrano l'Unità di controllo assemblata (tutti i fili sono stati sostituiti da piste nella versione definitiva del circuito).
Infine le figure 26 e 27 rappresentano la Scheda display lato componenti e lato saldature e la figura 28 le interconnessioni tra i circuiti, i collegamenti del commutatore Sw2 per la selezione dei tempi ed il connettore RS232 (connettore 1).

TARATURA

La taratura non è particolarmente complicata e consiste, per ogni scheda telecamera, nel regolare il guadagno dello stadio composto da U1 (ruotare il trimmer V1 in funzione della sensibilità desiderata) e nella regolazione di V2 in modo che su P3 out ci siano esattamente 1.5V (centro elettrico della finestra). Entrambe le regolazione vanno effettuate dopo avere inserito le schede nell'unità di controllo, senza collegare la telecamera al circuito (altrimenti la variazione del segnale renderebbe difficoltoso tarare il valore preciso di tensione).

REPERIBILITÀ DEI COMPONENTI

Alcuni dei componenti utilizzati nel progetto possono essere di difficile reperibilità (MAX764,

Figura 25: *Unità di controllo lato saldature*

Figura 26: *Scheda display lato componenti*

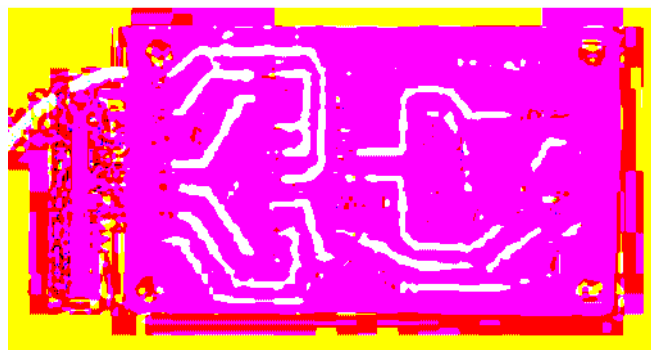



Figura 27: *Scheda display lato saldature*

MAX4315) presso i normali distributori di materiale elettronico, specialmente se la richiesta è limitata a pochi pezzi.
In alternativa consiglio di contattare il distributore Maxim più vicino (l'elenco si trova sul sito www.maxim-ic.com); eventualmente si può anche richiedere una campionatura gratuita (il discorso vale ovviamente solo per 2-3 pezzi) come free sample sempre sul sito della Maxim.
Tutti gli altri componenti sono facilmente reperibili.

SITI WEB UTILI

www.maxim-ic.com :è il sito della Maxim componenti; moltissime circuiti integrati innovativi e decisamente interessanti.
www.microchip.com: tutto quello che occorre per sviluppare i programmi sui PIC.
www.visual-basic.it: sito interessante di programmazione Visual Basic completo e in italiano.
www.active-x.com: se serve un active-X per VB6 qui c'è di sicuro.
www.iocomp.com: active-X per pannelli professionali; è possibile scaricare una demo perfettamente funzionante (anche se dura solo 10 minuti).

Figura 28: *Connessioni tra i circuiti*

CONCLUSIONI

Il circuito funziona ininterrottamente da diverso tempo e non ha causato falsi allarmi o instabilità particolari, tuttavia si tratta pur sempre di un circuito sperimentale suscettibile di migliorie. Ritengo inoltre che il circuito possa costituire anche la base per migliorare il software, per svilupparne di nuovo o anche solo per impiegare parti del progetto per realizzare circuiti di acquisizione dati tipo voltmetri o termometri scambiando dati con il protocollo sopra illustrato.
I files sorgente assembler e Visual Basic ed il pacchetto di installazione possono essere scaricati direttamente dal sito Internet di Fare Elettronica.
Detto ciò auguro buon divertimento a tutti.

PANNELLO LUMINOSO A MATRICE DI LED

seconda parte

di Luca Calore
l.calore@farelettronica.com

Nella puntata precedente vi ho presentato questo progetto illustrandone caratteristiche e pregi.
In questa seconda parte vi guiderò passo passo nella costruzione per terminare la puntata con la prima accensione...

Prima di partire con la spiegazione desidero ricordarvi che il progetto si compone di quattro schede moduli e di una scheda controlli connesse tra loro attraverso dei connettori per formare un pannello come indicato nella figura 1.
Vi consiglio di realizzare le cinque schede necessarie con la tecnica della fotoincisione su doppia faccia facendo attenzione a rifilare correttamente i bordi per evitare problemi nell'assemblaggio sul pannello. La dicitura LS indica il lato saldature mentre la dicitura LC indica il lato componenti.

REALIZZAZIONE PRATICA DEI MODULI

Saldare i componenti nelle schede in esame non è difficile ma bisogna prestare molta attenzione alla qualità delle saldature che effettuerete. Vi consiglio vivamente controllare punto per punto il vostro operato, armati di tester per evitare spiacevoli imprevisti che posso pregiudicare il coretto funzionamento di tutti i LED.
Vi consiglio di iniziare saldando i molti fori passanti presenti su ogni scheda introducendo un pezzo di reoforo refuso da altri lavori saldandolo sia nel lato componenti che nel lato saldature.
I LED devono essere saldati orientati con il catodo che segue la lettera **C** stampata nel master tenendoli a circa 5 mm dalla superficie della scheda.

REALIZZAZIONE PRATICA DELLA SCHEDA CONTROLLI

La scheda dei controlli è realizzata sempre in doppia faccia con la particolarità di avere i componenti più alti montati sul lato rame (con le saldature sul lato componenti) come indicato nel piano di montaggio.
Questi componenti sono:

- Connettore Telefonico **JP1**
- Morsettiere **J4**, **J6**, **J8**
- Jumper **JP2**, **JP3**
- Batteria Tampone **BT1**
- Connettore Seriale **RS1**

COSTRUZIONE DELLA SCATOLA

La scatola è stata progettata per proteggere la matrice da urti che

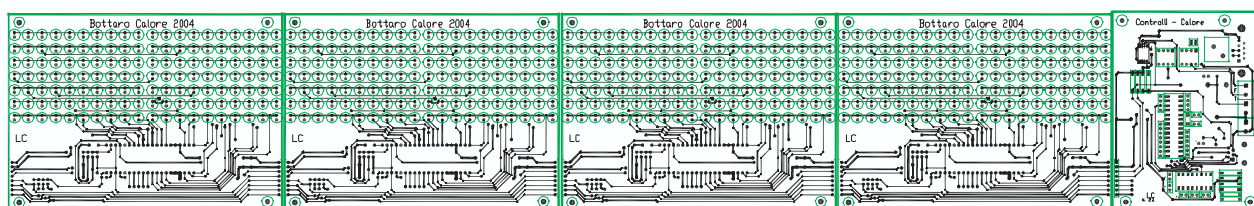



Figura 1: *Disegno del pannello completo*

la possono danneggiare fornendole allo stesso tempo un design più completo.
Il lamierino 15/10 conferisce al profilo la necessaria resistenza agli urti e può essere tranquillamente colorato con una comune bomboletta di colore acrilico.

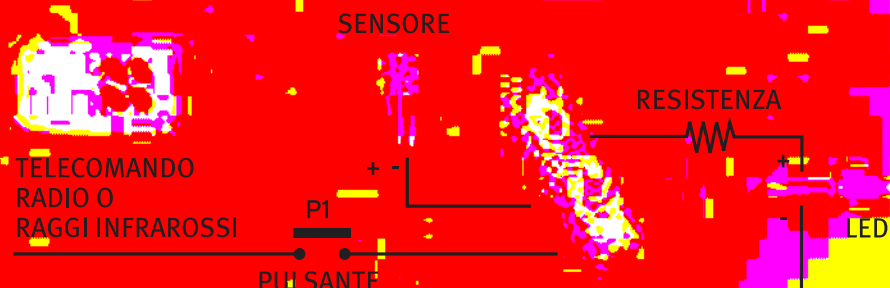
MONTAGGIO SULLA PIASTRA

La piastra in lamierino 15/10, illustrata nella figura 7, ha la funzione di vincolare l'insieme delle schede in modo da garantire in ogni condizione il miglior funzionamento.
Sulla piastra vanno praticati due serie di fori. I più grandi, da effettuare con un diametro di 20 mm, permettono di operare sui jumper senza dover smontare tutta la struttura.
I fori di diametro 2.5 mm sono le sedi per montare i distanziatori sui quali vengono appoggiate e fissate con viti le cinque schede che compongono il progetto.

I CONNETTORI PER LE SCHEDE

I moduli e la scheda controlli alloggiati nelle rispettive sedi distano 5 mm. Per congiungere ogni scheda sono necessarie quattro copie del connettore, come riportato in figura 8.
Per la realizzazione potete recuperare dal vostro laboratorio dei pezzi di basetta millefiori e su di essa posizionare otto file di strip (tulipano femmina) tagliati a gruppi di tre. Con un saldatore create le piste in figura connettendo i tre reofori di ogni gruppo di connettori.

SETUP HARDWARE

Prima di iniziare a programmare la matrice costruita sono necessarie alcune operazioni per consentire al microcontrollore di poter comunicare con tutte le periferiche installate.
Muniti di quattro jumper girate la piastra sotto sopra tendo sulla vostra sinistra la scheda controlli, individuato il blocco dei jumper, evidenziato in figura 9, inserite i jumper a partire dalla sinistra nel seguente ordine: 3° jumper [Gnd], 4° jumper [Vcc], 1° jumper [SDA], 2° jumper [SCL].

Completando questa operazione

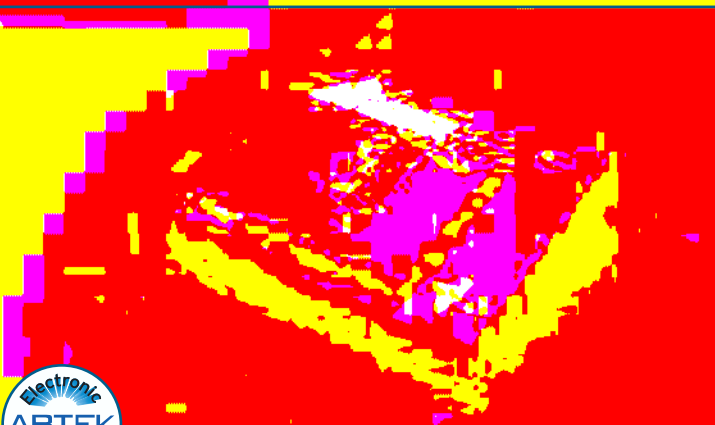
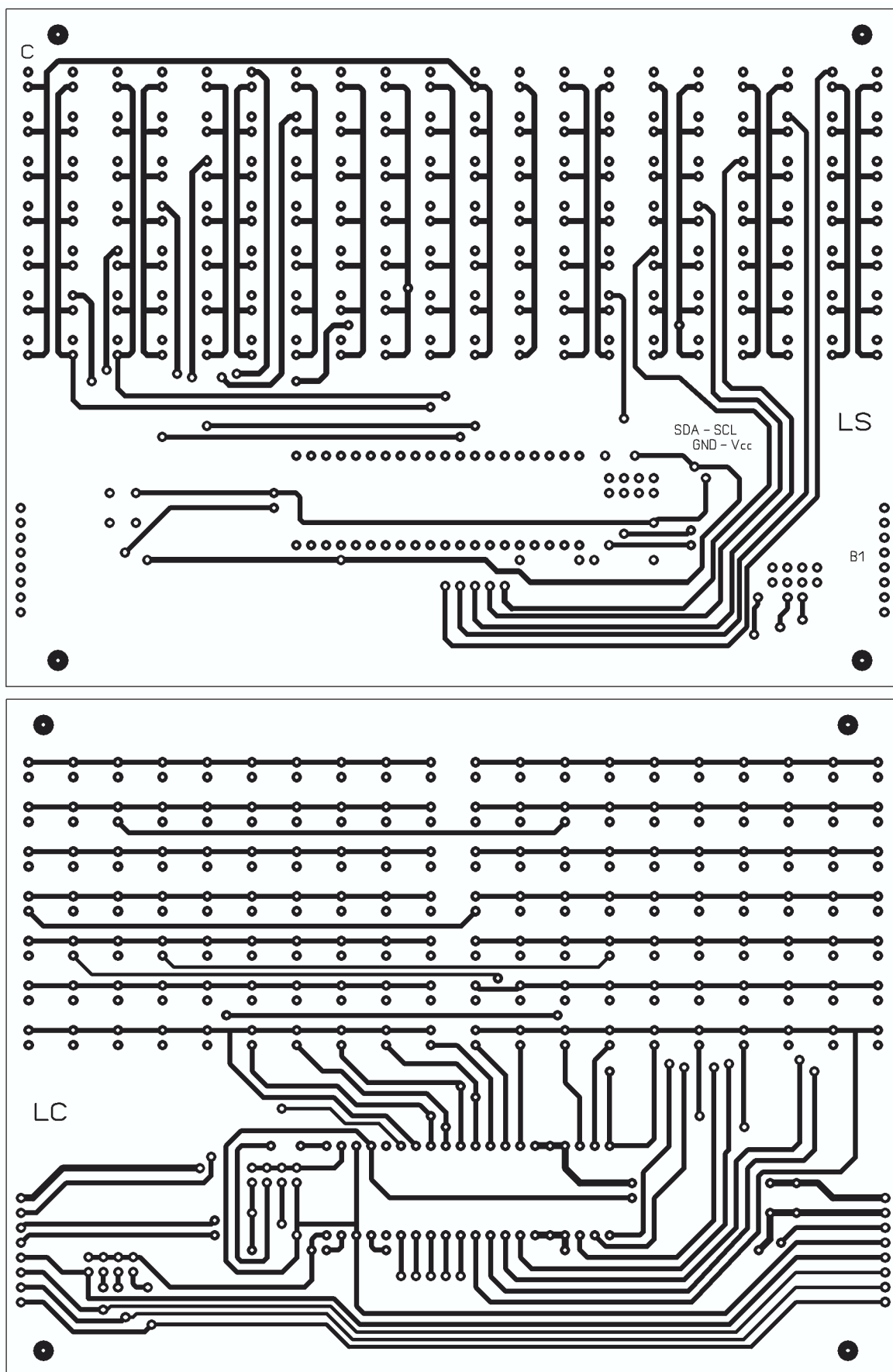



Figura 2: *Circuito stampato in scala 1:1 del modulo LED*

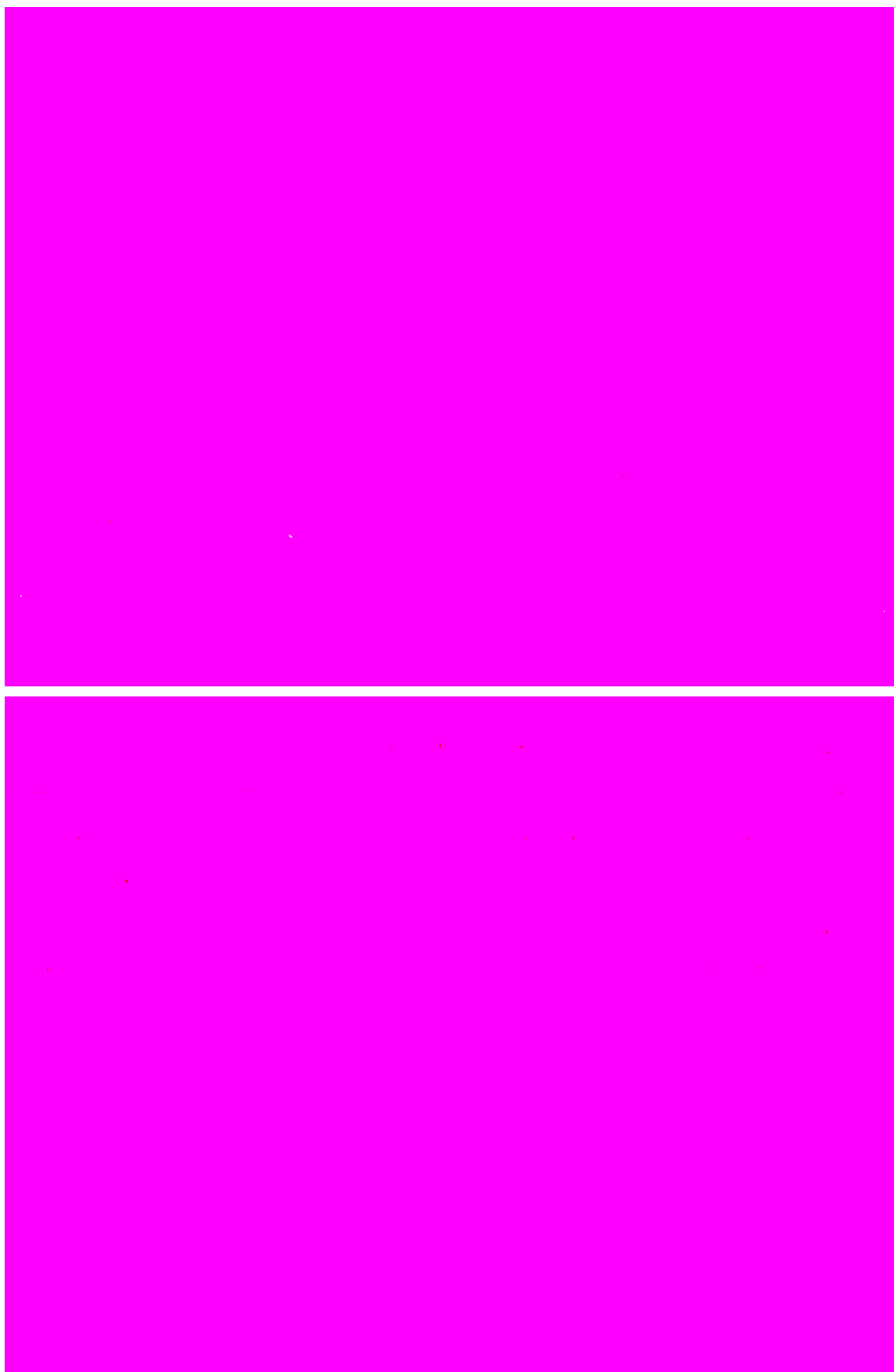



Figura 3: *Piano di montaggio del modulo LED*

Elenco componenti – Schede Moduli		
Quantità	**Sigla**	**Valore**
1 x 4	R1	4,7 KΩ 1/4 W (± 1% ove possibile)
1 x 4	R2	68 KΩ 1/4 W (± 1% ove possibile)
2 x 4	C1,C2	100 nF ceramico
1 x 4	C3	15 pF ceramico
140 x 4	LD1÷LD140	LED rossi 5mm (preferibilmente alta luminosità)
1 x 4	U1	Maxim MAX6953
8 x 4	JP1÷JP8	Connettori IDC passo 2,54 mm (coppia) e ponticelli
16 x 4	J1, J2	Connettori IDC con passo 2,54 mm maschi

avete settato univocamente gli indirizzi delle quattro schede dei moduli lungo il bus I2C.
Con un altro jumper dovete chiudere un ponticello, nella posizione evidenziata in figura 10, in un solo modulo. In questo modo il microcontrollore prende quel modulo come riferimento per la temporizzazione delle funzioni ed in particolare per la gestione del lampeggio e del lampeggio alternato. Volendo approfondire il significato di questo ponticello potete consultare il datasheet del MAX6953 cercando la documentazione riguardo al pin 20.

CARICAMENTO DELLA BATTERIA TAMPONE

Il caricamento della batteria tampone, che funge da alimentazione di backup per il RTC (Real Time Clock) inserito nella scheda controlli, può essere fatto aprendo il jumper J2 e collegando al connettore, evidenziato nella figura 11, (rispettando le polarità) un comune caricabatteria per accumulatori Ni-Cd.
L'operazione è molto semplice ma molto delicata, infatti, il RTC è estremamente sensibile alle sovratensioni ai capi della batteria.

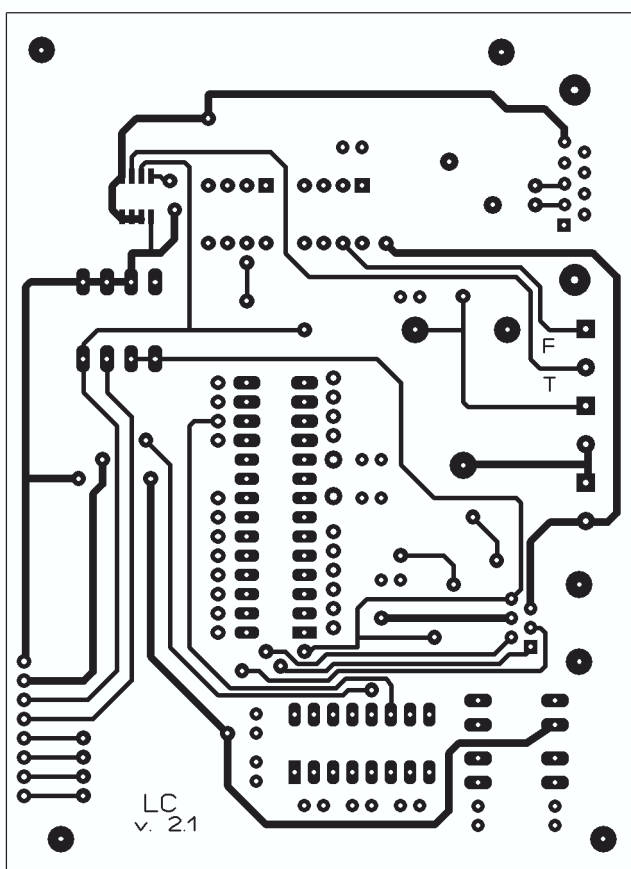
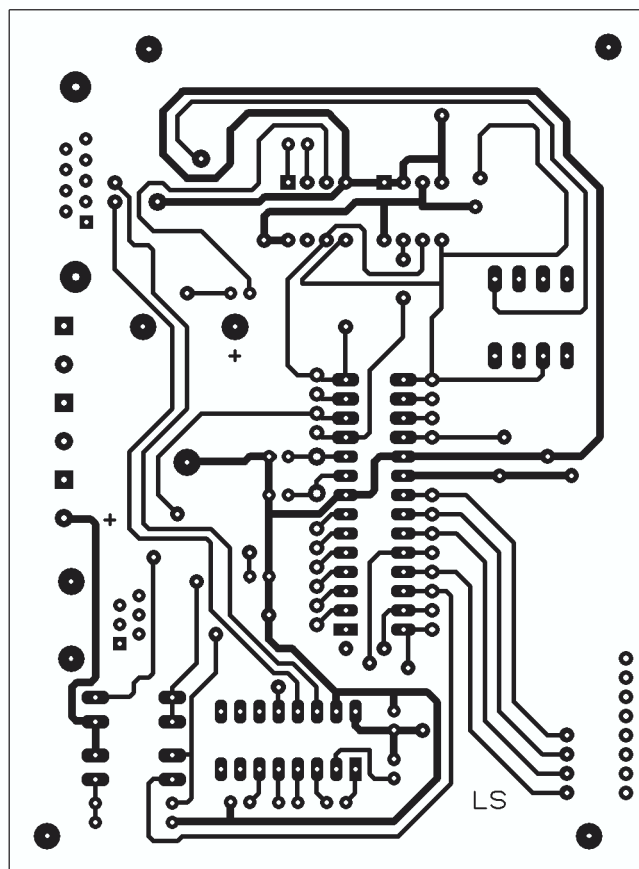



Figura 4: Circuito stampato in scala 1:1 della scheda controlli

Come riportato nel datasheet di questo componente, la tensione della batteria tampone non deve superare i 3.5 V, vi consiglio quindi di controllarne periodicamente la tensione ai capi durante la rica-

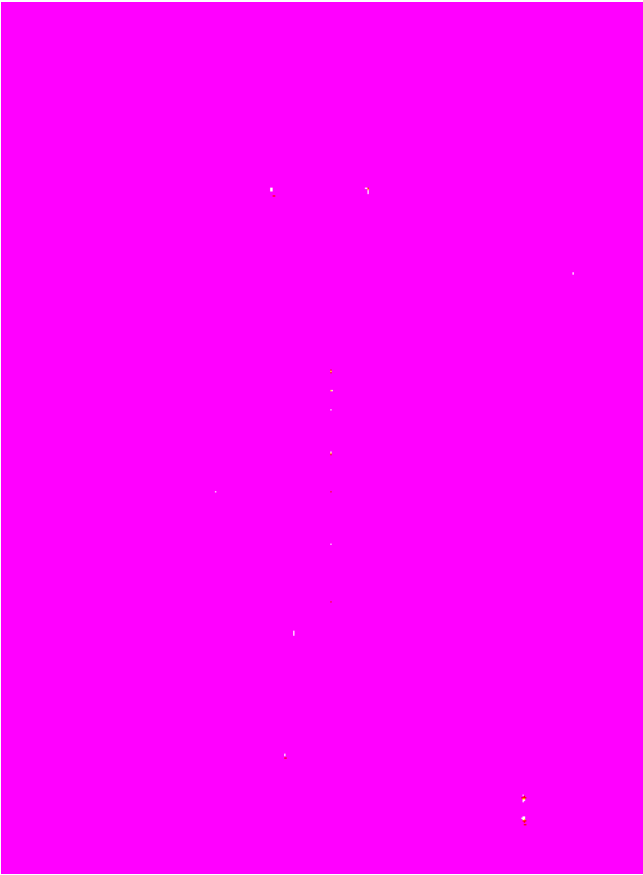
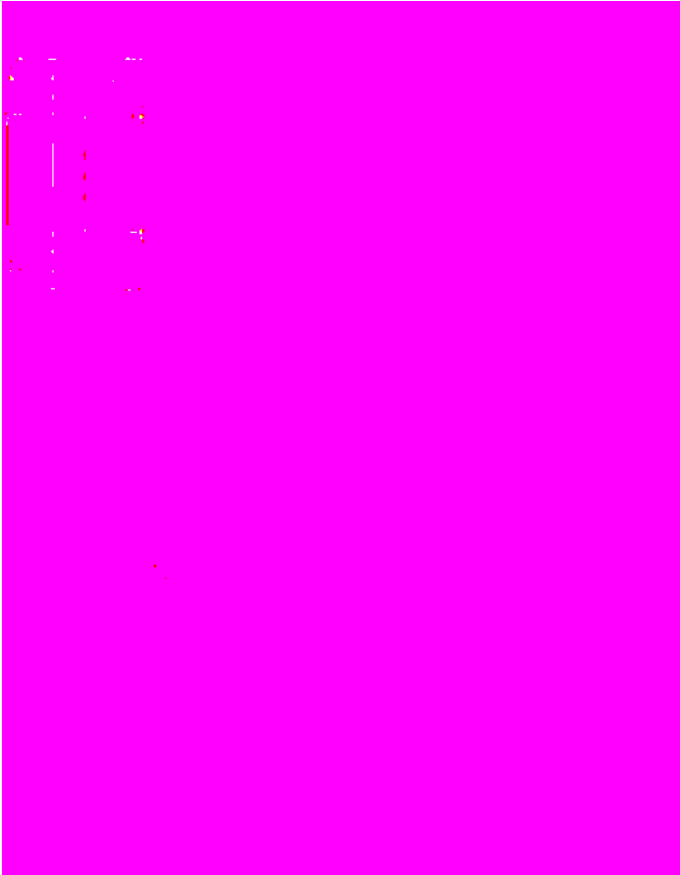



Figura 5: *Piano di montaggio della scheda controlli*

Elenco componenti – Scheda Controlli					
Quantità	**Sigla**	**Valore**	**Quantità**	**Sigla**	**Valore**
2	R1, R2	10 KΩ 1/4 W	1	Y1	Risuonatore ceramico 4 Mhz
2	R3, R4	33 KΩ 1/4 W	1	Y2	Quarzo 32.768 KHz
2	R6, R8	470 Ω 1/4 W	1	BT1	Batteria tampone 3.6 V
2	C1, C2	Condensatori 22pF	1	JP1	Connettore telefonico
5	C3÷C7	Condensatori elettrolitico 1μF 50V	2	JP2, JP3	Connettori IDC passo 2,54 mm (coppia) e ponticelli
2	D1, D2	Diodi 1N4148	8	J2	Connettori IDC con passo 2,54 mm maschi
1	U1	Microchip PIC16F86	22	J7, J3, J5, J9	Connettori strip-line femmine
1	U2	Microchip 24AA515	3	J4, J6, J8	Connettori 2 poli passo 5 mm
1	U3	Dallas Semiconductor DS1207	1	RS1	Connettore DB9 Seriale
1	U4	Dallas Semiconductor DS1631	1	S1	Pulsante NA da cs
1	U5	Maxim MAX232			

rica. Completata l'operazione inserite il jumper J2 prima rimosso.
La batteria ricaricata non dovrebbe scaricarsi prima di dieci anni di servizio.

SETUP SOFTWARE

Prima di caricare il firmware all'interno del microcontrollore vi consiglio di controllare con attenzione se le linee distribuite lungo la matrice, attraverso i connettori, risultano condurre e sono isolate reciprocamente. Ponete molta attenzione alle prime due linee che trasportano l'alimentazione, soffermandovi anche a controllare se risultano isolate rispetto alla piastra.
Ora potete collegare il vostro alimentatore al connettore evidenziato in figura 12 rispettando le polarità, in modo da alimentare la matrice. Vi ricordo velocemente che è necessario un alimentatore che eroghi almeno 5V – 1.5A.

Il microcontrollore può essere programmato con qualsiasi programmatore usando i file sorgenti (dopo averli compilati) o direttamente il file Hex che è possibile scaricare dal sito Web di Fare Elettronica.
La compilazione dei file sorgente può avvenire usando Mplab con la seguente configurazione:

```
CP1:CP0:        OFF
DEBUG:          ON
WRT:            OFF
CPD:            OFF
LVP:            OFF
BODEN:          Disabled
PWRTE:          Enabled
WDTE:           Disabled
FOSC1:FOSC0:    XT
Case sentive:   OFF
Radix:          HEX
```

Elenco componenti Connettori e scatola	
Quantità	**Valore**
96	Connettori IDC con passo 2,54 mm femmina tagliati tre a tre
20	Distanziatori 25 mm diametro 3 mm

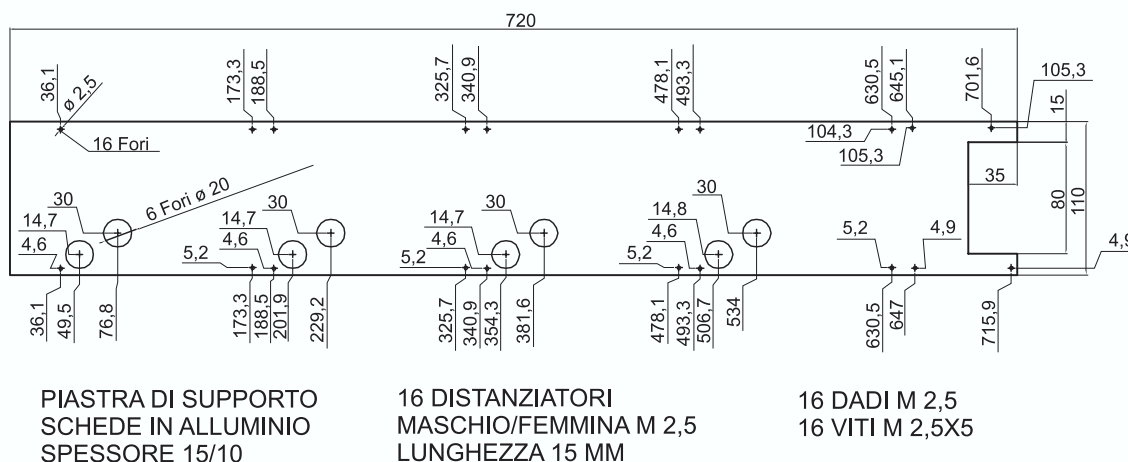



Figura 6: Disegno meccanico della scatola

Figura 7: Disegno meccanico della piastra

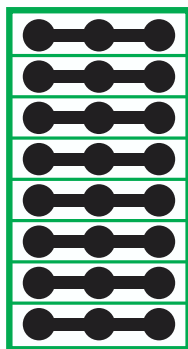



Figura 8: *Schema del connettore da realizzare in quattro copie*

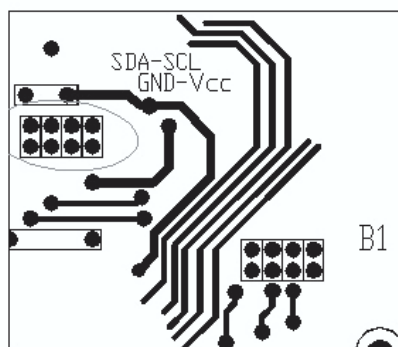



Figura 9: *Posizione Jumper Indirizzo*

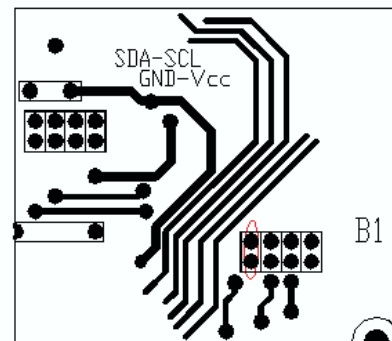



Figura 10: *Posizione Jumper Temporizzazione*

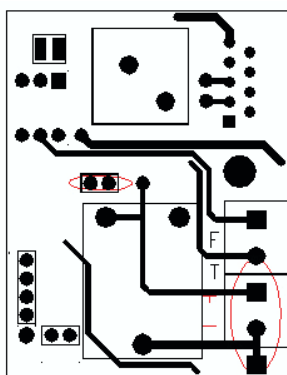



Figura 11: *Posizione Jumper e Connettore per la ricarica della batteria tampone*

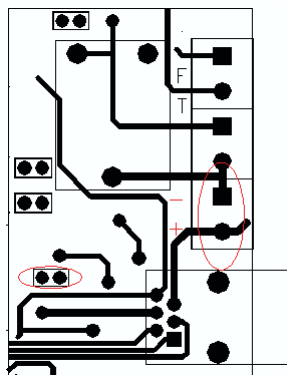



Figura 12: *Posizione del connettore alimentazione*

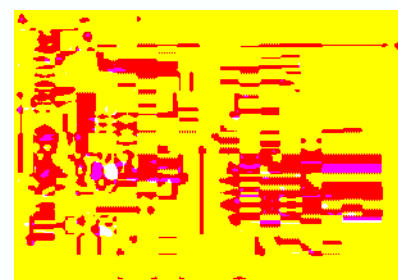



Figura 13: *Schermata Opzioni/Configurazione del programma di gestione*

Installate poi sul vostro PC il programma di gestione della matrice, che potete scaricare dal sito Web di Fare Elettronica, in modo da consentirne la programmazione.

AVVIO E TEST DI FUNZIONAMENTO

La prima volta che eseguite il programma che controlla la matrice di LED è necessario che andiate a chiudere il jumper JP3 evidenziato in figura 9, in questo modo sul display apparirà la dicitura "tasto bloccato" che disabilita la lettura dei dati dalla EEprom (perché non sarebbero validi) ma lascia attive le procedure di controllo remoto.
Ora potrete eseguire i due test software per controllare se tutto funziona. Aprite il programma di gestione ed entrate nella schermata rappresentata in figura 13. Nell'assicuravi che la matrice è connessa alla giusta porta seriale spuntate le caselle riguardanti i test.
Se avete completato il montaggio con successo dovrete verificare che tutti i led si accendino per tre secondi e che per i successivi tre secondi venga visualizzata l'immagine di figura 14, la quale indica un corretto allineamento dei moduli.
Se i test hanno avuto successo potete caricare la scena test (dal file test.sce), della schermata "modifica scene", cliccando il bottone "invia scene" (figura 15).
Per questa puntata ho concluso, nel darvi appuntamento alla prossima, vi anticipo che troverete la guida al programma di gestione e la descrizione del funzionamento del firmware.

Figura 14: *Risultato Test Posizione*

Figura 15: *Schermata per l'invio delle scene*

CAMPAGNA ABBONAMENTI 2004-2005

Abbonarsi a Fare Elettronica significa ricevere, comodamente ogni mese a casa tua, tante idee e consigli per rendere il tuo **hobby** una vera **passione.**

Perché Fare Elettronica si diverte solo quando ti diverti tu.

Abbonandoti potrai ricevere la rivista ad un prezzo molto interessante rispetto a quello di copertina. Pagherai infatti solo € 39 invece di € 51 con un risparmio di ben € 12 oltre a ricevere uno sconto del 10% su tutti i libri del catalogo SANDIT (più di 800 titoli).

ABBONARSI A FARE ELETTRONICA CONVIENE.

Abbonarsi subito conviene ancora di più.

Diverse possibilità di abbonamento:

Standard: Il tuo abbonamento personale o aziendale al costo di € 39

Regalo: Se sei già abbonato e vuoi regalare un nuovo abbonamento ad un amico, lo pagherai solo € 35 (10% di sconto) comunicando il tuo codice

Scuole: Riservato a scuole ed università; ordinando quattro abbonamenti ne riceverai uno in omaggio, pagherai quindi € 156 anzichè € 195 (20% di sconto)

Come abbonarsi:

Per Posta: scrivere a INWARE Edizioni - Via Cadorna, 27 - 20032 Cormano (MI)

Per Telefono: al numero +39 02.66504794

Per Fax: al numero +39 02.66508225

Via Internet: sul sito www.farelettronica.com alla pagina campagna abbonamenti

(è possibile pagare con bollettino postale, bonifico bancario e carta di credito, maggiori dettagli in ultima pagina)

Il Bundle di Fare Elettronica è una combinazione di prodotti ad un prezzo scontatissimo.

NON LASCIARTI SCAPPARE IL TUO PREFERITO!

Elenco dei prodotti attualmente disponibili in Bundle	Prezzo normale Cad.
❑ Fare Elettronica N. 212 Febbraio 2003	€ 7,50
❑ Fare Elettronica N. 213/214 Marzo/Aprile 2003 – Numero Doppio	€ 9,00
❑ Fare Elettronica N. 219 Settembre 2003	€ 7,50
❑ Fare Elettronica N. 220/221 Ottobre/Novembre 2003 – Numero Doppio	€ 9,00
❑ Fare Elettronica N. 222 Dicembre 2003	€ 7,50
❑ Fare Elettronica N. 223 Gennaio 2004	€ 7,50
❑ CD-ROM “Annata Fare Elettronica 2003”	€ 25,00
❑ Software Abacom “sPlan Disegno schemi elettrici”	€ 46,80
❑ Software Abacom “Sprint Disegno circuiti stampati”	€ 46,80
❑ Software Abacom “Front Designer Disegno pannelli frontali”	€ 46,80

Elenco dei Bundles disponibili	Prezzo Bundle Cad.
❑ Bundle “A” : 2 Arretrati (qualsiasi) al prezzo speciale di	€ 12,00
❑ Bundle “B” : 3 Arretrati (qualsiasi) al prezzo speciale di	€ 15,00
❑ Bundle “C” : 1 Arretrato (qualsiasi) e il CD-ROM ”Annata FE 2003” al prezzo speciale di	€ 30,00
❑ Bundle “D” : 1 Arretrato (qualsiasi) e 1 Software Abacom (qualsiasi) al prezzo speciale di	€ 50,00
❑ Bundle “E” : 2 Software Abacom (qualsiasi) al prezzo speciale di	€ 84,00

Indicare il Bundle e i prodotti prescelti mettendo una croce sulle rispettive caselle.

Per ordinare il tuo Bundle utilizza questa pagina inviandola via Fax al n. 02.66508225, oppure puoi ordinarlo online all'indirizzo www.farelettronica.com/bundle

Tutti i Bundle sono comprensivi di IVA e spese di spedizione. Il pagamento è anticipato, e può essere effettuato tramite:

❑ **Bollettino postale** Utilizzare il **C/C N. 22790232** intestato ad **Inware srl**, indicando nella causale **“Ordine Bundle”**

❑ **Bonifico bancario** Appoggiarlo sulla banca: **Poste Italiane - CIN: Z - ABI: 07601 - CAB: 01600 - C/C: 000022790232**

❑ **Carta di credito** Titolare: ______ Numero: ______ Scadenza: __ / __ VISA MasterCard

Cognome ______ Nome ______

Azienda ______

Via ______ CAP ______ Città ______ Prov. ______

Tel. ______ Fax ______ email ______

SMARTCARD quarta parte

UNIREADER: PROGETTO DI UN LETTORE DI SMARTCARD

di Giuseppe Modugno
gppe.modugno@libero.it

In questo quarto articolo del nostro tutorial, verrà progettato un lettore universale di smartcard compatibile sia con le carte a memoria, come le SLE4442, sia con quelle a microprocessore, come le SIM dei cellulari GSM. Il cuore di UniReader è un microcontrollore PIC 16F628 che permette di utilizzare il lettore sia in connessione con un PC, sia in modalità standalone.

RIPRENDIAMO LA NOSTRA PARTITA A CARTE

Cosa ci siamo detti negli articoli precedenti

Prima di poter affrontare la progettazione e la realizzazione di un lettore di smartcard, è necessario conoscere in dettaglio questi oggetti in tutti i loro aspetti. Per questo motivo, potete trovare, nei precedenti articoli, una trattazione teorica sulle smartcard che indaga sulle varie tipologie e ne discute i campi di utilizzo attuali e futuri. Inoltre, potete trovare un articolo sulle smartcard a memoria ed uno sulle smartcard a microprocessore, i due tipi in cui si classificano tutte le smartcard, entrambi compatibili con UniReader. Non mi limiterò semplicemente a mostrare uno schema elettrico e il suo sbroglio con le solite istruzioni per il montaggio, ma cercherò di descrivere in dettaglio tutte le fasi di progettazione giustificando le varie scelte. Per comprendere al meglio il contenuto di questo articolo, consiglio al lettore di leggere anche i precedenti articoli del tutorial.

Cos'è un lettore di smartcard

Prima di lanciarci nella progettazione, è necessario riflettere sull'oggetto che dobbiamo realizzare e che abbiamo chiamato lettore universale di smartcard. L'attributo universale sta ad indicare che esso deve essere compatibile sia con le smartcard a memoria, sia con le smartcard a microprocessore: qualunque sia la carta in nostro possesso, il lettore dovrà essere in grado di gestirla correttamente.
Il termine lettore, invece, è sicuramente da commentare in modo più approfondito. Se domandiamo ad un profano quali compiti deve eseguire un lettore di smartcard, sicuramente ci risponderà che deve essere in grado di leggere i dati memorizzati nella carta e presentarli all'utente in varia forma (per

esempio, su carta, su display, ecc) oppure trasmetterli ad un altro dispositivo elettronico (per esempio, ad un computer). Chi ci ha seguito con attenzione sin dalla prima parte del tutorial, può subito capire come questa immagine sia alquanto riduttiva. Infatti, un lettore di smartcard non si limita a leggere ma effettua sulla carta una serie di altre operazioni più o meno complesse che possono andare dalla lettura dei dati presenti sulla carta all'autenticazione dell'utente, dalla modifica dei dati (scrittura) all'esecuzione di algoritmi di cifratura complessi. Per quello che abbiamo detto, il termine "lettore di smartcard" appare decisamente restrittivo. Sarebbe meglio parlare a tal proposito di dispositivo di interfacciamento con la smartcard o, dall'inglese, *interface device* (**IFD**) ma, pur nell'imprecisione, nel seguito di questo articolo faremo riferimento al termine *lettore di smartcard* poiché più comunemente usato.
A prima vista può sembrare che un dispositivo di questo tipo, che può compiere diverse operazioni, anche complesse, con una qualsiasi smartcard, sia di difficile realizzazione, soprattutto per un hobbista. In realtà, come vedremo nel corso dell'articolo, tutte queste operazioni si riducono, al livello più basso, alla trasmissione e ricezione di comandi, formati da una successione più o meno lunga di byte verso e dalla smartcard. Il particolare comando determina l'operazione, anche complessa, che l'utente, tramite il lettore, vuole far eseguire alla smartcard.

Da queste considerazioni ci si può convincere di come un lettore sia semplicemente un dispositivo di interfaccia che converte le "volontà" dell'utente o del sistema in opportuni comandi da trasmettere o dati da ricevere dalla smartcard, mediante un opportuno protocollo.

PROGETTAZIONE

Ovviamente, tralasciamo la parte che dovrebbe precedere qualsiasi progetto e che corrisponde alla documentazione e allo studio, poiché coincide con i nostri precedenti articoli.
Partiamo, quindi, da una panoramica generale del nostro dispositivo, cioè da un'analisi attenta del suo schema a blocchi.

Schema a blocchi

Come abbiamo già detto, il nostro lettore di smartcard è un dispositivo di interfacciamento che si preoccupa di convertire le "volontà" dell'utente o del sistema in comandi da trasmettere e ricevere dalla smartcard. Questo ruolo di "intermediazione" è facilmente comprensibile guardando lo schema in figura 1.
Da una parte c'è l'utente dell'applicazione che può essere rappresentato dalla persona al Bancomat oppure dal telefonino GSM. In quest'ottica, il termine UTENTE utilizzato in Figura 1 è un concetto molto più allargato rispetto alla persona fisica. Dall'altra parte è presente la smartcard, dove sono memorizzati i dati necessari per l'esecuzione dell'operazione (per esempio, numero di conto corrente o crediti) oppure dove è possibile effettuare operazioni di cifratura per l'autenticazione al sistema (per esempio, alla rete GSM del nostro provider). Al centro è presente il LETTORE di smartcard che si preoccupa di convertire il flusso di operazioni provenienti dall'UTENTE in comandi da trasmettere alla SMARTCARD e viceversa.
L'UTENTE può comunicare con il lettore in entrambe le direzioni e in diversi modi, a seconda dell'applicazione. Il lettore può gestire il flusso in input (UTENTE ‡ LETTORE) mediante una tastiera, una connessione seriale RS232 o USB, ecc. Allo stesso modo, in output (LETTORE > UTENTE) il lettore può gestire dei led, un display LCD, una connessione RS232 o USB, dei relé, ecc. Se, per esempio, l'ap-

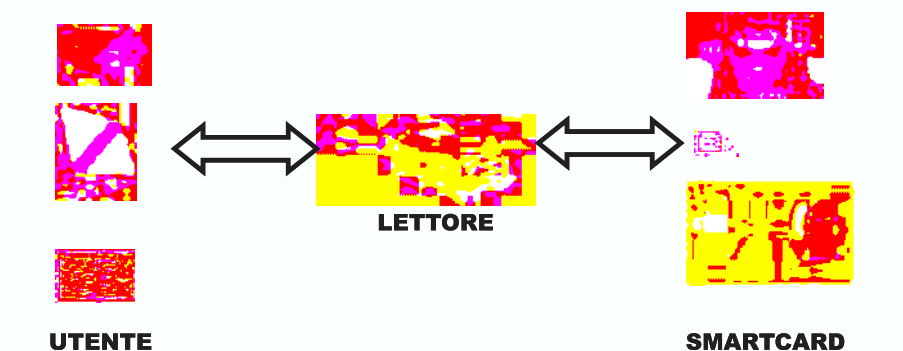



Figura 1: Il lettore come dispositivo d'interfacciamento

plicazione prevede l'apertura di un cancello all'inserimento di una smartcard valida, utilizzeremo un relé pilotato dal lettore che azionerà l'apricancello ad autenticazione correttamente avvenuta.
Per quanto riguarda il lato SMARTCARD, l'implementazione è facilitata dal fatto che sia le smartcard a memoria che le smartcard a microprocessore standard hanno lo stesso formato e la stessa disposizione dei contatti: questo ci permette di utilizzare un unico connettore ISO7816 da interfacciare al lettore.
Dal lato UTENTE, invece, facciamo la scelta di implementare nel nostro prototipo soltanto una connessione seriale RS232 con un eventuale PC. In questo modo, mediante un semplice software per computer, sarà possibile gestire la comunicazione con il lettore in entrambe le direzioni e, quindi, con la smartcard. Per mostrare come sia possibile far funzionare il lettore in modalità stand-alone, è previsto un led che può significare l'abilitazione all'utilizzazione della risorsa nel caso di accesso consentito al titolare della smartcard.

La scelta del microcontrollore

Precisiamo subito che la presenza di un microcontrollore in un lettore di smartcard non è necessaria. Esistono dei semplici lettori, chiamati generalmente dumb (stupidi), in cui i contatti della smartcard sono collegati ai pin di una seriale RS232 mediante una semplice traslazione di livello (ricordiamo che la smartcard è compatibile con logica a 5V, mentre la seriale RS232 prevede segnali di circa –10V e +10V). Ciò è possibile poiché, come ricorderà chi ha letto l'articolo sulle smartcard a microprocessore, il protocollo T=0 è molto simile al protocollo RS232 con 8 bit di dati, 1 bit di parità, 2 bit di stop alla velocità di 9600 bit/s. Tutta la logica di funzionamento viene quindi completamente spostata nel software del computer. Anche le smartcard a memoria possono essere gestite direttamente dalla RS232 di un PC senza un microcontrollore, utilizzando alcuni pin della seriale (DTR, RTS, ecc) per pilotare direttamente i segnali CLK, RST e I/O.
La domanda è d'obbligo: perché abbiamo utilizzato un microcontrollore nell'UniReader? Per alcuni motivi che andiamo subito a commentare.

- La presenza di un microcontrollore ci permette di utilizzare il lettore anche in modalità stand-alone, senza alcuna connessione con un PC. Il software di gestione non sarà in esecuzione nel PC, ma nel microcontrollore.
- Il protocollo T=0, seppur molto simile al protocollo utilizzato nella RS232, si discosta in alcuni punti da questo. Per esempio, nel protocollo T=0, utilizzato dalle smartcard a microprocessore, è previsto un controllo di parità "al volo" che è impossibile implementare utilizzando la sola RS232 (per maggiori dettagli si veda l'articolo precedente sulle smartcard a microprocessore).
- La presenza di un microcontrollore ci permette di riutilizzare l'hardware anche per altri tipi di smartcard e/o protocolli di comunicazione anche molto diversi dal protocollo RS232. In questo caso, basta cambiare opportunamente il firmware del microcontrollore per gestire tali particolari situazioni.
- Mediante il microcontrollore è possibile gestire della circuiteria elettronica aggiuntiva che potrebbe essere impossibile pilotare tramite la sola connessione seriale (si pensi, per esempio, all'eccitazione di un relè in funzione della carta inserita nel lettore oppure ad un display LCD).

Dopo aver giustificato l'utilizzo di un microcontrollore, passiamo alla sua scelta. Come abbiamo detto, si utilizzerà il PIC 16F628 della Microchip che ci è sembrato una buona soluzione per tanti motivi.
Innanzitutto, è un chip molto diffuso nei negozi di elettronica ad un prezzo sicuramente accessibile alla maggior parte degli hobbisti. Inoltre, per la memorizzazione del firmware, è dotato di una memoria Flash da 2048 istruzioni che permette la riprogrammazione in caso di errori o di aggiornamenti del programma. È dotato di un oscillatore interno a 4MHz, a differenza del suo "fratello più piccolo" 16F84, che semplifica ulteriormente la complessità del circuito. Internamente ha una porta USART che permette l'interconnessione con una RS232 di un PC in modo molto semplice sia a

livello hardware, ma soprattutto a livello di programmazione (il 16F84 non è dotato di porta USART).
Inutile puntare su microcontrollori più evoluti, poiché ciò non giustificherebbe l'aumento del prezzo e della complessità in relazione ai nostri scopi. È ovvio che, a seconda delle necessità, sarà possibile riprogettare il lettore utilizzando un qualsiasi altro microcontrollore.
Scegliendo, invece, il più piccolo 16F84, avremmo dovuto rinunciare a metà della memoria Flash, con la conseguenza di non poter creare programmi molto complessi; alla porta USART, con la maggiore complessità di programmazione relativa alla gestione della connessione RS232 e quindi ad un maggior aumento del codice, già penalizzato da una memoria ridotta; all'oscillatore interno a 4MHz, con la conseguenza di complicare il circuito.
Pensiamo che il minor costo del 16F84 non giustifichi tutti questi svantaggi, anche perché la differenza di prezzo è, in molti casi, irrisoria o addirittura assente.

Schema elettrico

In figura 2 è mostrato lo schema elettrico del nostro lettore di smartcard basato su PIC 16F628.
L'alimentazione dell'intero circuito viene fornita mediante il connettore J1 a due contatti, a cui è possibile applicare una tensione continua di almeno 8V. Per esempio, è possibile collegare direttamente una batteria da 9V, magari sostituendo il connettore J1 con il classico connettore per tali batterie, oppure è possibile utilizzare un qualsiasi alimentatore da muro che abbia una

Figura 2: *Schema elettrico*

uscita continua di almeno 8V (generalmente 9V o 12V). Non è necessario che tale tensione sia regolata, poiché essa viene stabilizzata mediante un regolatore lineare 7805 (IC1) che genera il livello di 5V utilizzato per l'alimentazione della smartcard, del microcontrollore e degli altri integrati.
Il diodo D1 ha il solo scopo di evitare problemi nel caso si colleghi il polo positivo e negativo dell'alimentazione nel verso opposto. Il led verde DL1 è collegato direttamente alla tensione di 5V, quindi si accende ogni volta che viene alimentato il circuito e può essere usato come test di presenza alimentazione.
L'interruttore S1 è utile per staccare l'alimentazione dal circuito. Il cuore del nostro lettore di smartcard è il microcontrollore PIC 16F628 (IC3). Esso verrà programmato in modo da utilizzare l'oscillatore interno a 4MHz, quindi i pin RA7/OSC1 e RA6/OSC2 sono usati come pin di input/output generici. Il pin MCLR non viene usato, quindi la funzionalità di reset dovrà essere disabilitata durante la programmazione, azzerando il flag relativo.
I pin RX e TX sono rispettivamente i pin di ricezione e trasmissione della porta USART che ci permette di gestire in modo semplice la programmazione del protocollo seriale. Lo standard RS232 prevede livelli positivi di circa +10V e livelli negativi di –10V, mentre la USART del microcontrollore gestisce segnali TTL 0-5V. Per questo motivo è necessario utilizzare un circuito traslatore di livello che converta i livelli TTL in RS232 e, viceversa, i livelli RS232 in livelli TTL. Il chip maggiormente usato per questo scopo è il MAX232 (IC2) prodotto dalla Maxim o uno dei suoi numerosissimi equivalenti di altri produttori, molto spesso compatibili anche con il pinout. Esso realizza questa conversione di livello mediante la tecnica definita "pompa di carica" che si serve di quattro condensatori esterni (C7, C8, C9, C10) per raddoppiare la tensione da 5V a 10V e creare la tensione negativa –10V. Il connettore RS232 utilizzato è il tipico D-SUB 9 pin femmina, in cui è usato il pin 2 per la trasmissione ed il pin 3 per la ricezione. Il connettore maschio a 9 poli generalmente utilizzato nei PC come porta seriale, utilizza, invece, il pin 3 per la trasmissione ed il pin 2 per la ricezione. Questo ci permette di collegare direttamente il lettore di smartcard alla porta seriale del computer mediante un cavo "pin to pin" maschio-femmina dove, cioè, il pin 1 del connettore maschio è collegato al pin 1 del connettore femmina, il pin 2 del maschio è collegato al pin 2 della femmina, ecc. In realtà, la nostra seriale utilizza soltanto tre conduttori (RX, TX e GND), quindi basterebbe un semplice cavo dove siano collegati i pin 2, 3 e 5. Gli altri eventuali conduttori non sono usati e potrebbero essere assenti nel cablaggio senza inficiare la corretta comunicazione.
Il connettore per smartcard (J3) utilizzato, prodotto dalla AMP, oltre agli 8 contatti ISO7816, presenta due contatti aggiuntivi (SW1 e SW2) che sono normalmente chiusi e si aprono quando viene inserita una carta. Tramite la linea SC_DETECT ed il pull-up R3, il microcontrollore può sapere se nel lettore è stata inserita una smartcard o meno. Esistono in commercio connettori ISO7816 senza questi due contatti oppure con contatti normalmente aperti: in questi casi basta modificare leggermente il firmware del microcontrollore per ignorare la linea SC_DETECT oppure per invertirne la logica di funzionamento. I pin RST, VPP e I/O sono direttamente collegati al microcontrollore ai pin RA2, RA1 e RA4 rispettivamente. I pin RST e VPP sono normalmente degli ingressi per la smartcard, quindi delle uscite per il microcontrollore. In questo prototipo la linea VPP è stata collegata ad un pin del microcontrollore, anche se difficilmente verrà gestita, in quanto generalmente non utilizzata o, nel caso in cui venga usata, spesso rappresenta una tensione maggiore di 5V che il microcontrollore non può fornire. Queste situazioni non possono essere gestite da UniReader, poiché molto diverse da caso a caso. Il pin I/O, invece, è una linea bidirezionale, quindi dovrà essere usata sia come input che come output dal microcontrollore: il pin RA4 ha questa caratteristica. In alcuni casi, come nelle smartcard a memoria SLE4442, il pin di I/O è un open-drain ed è quindi necessario utilizzare un pull-up (R4) per generare entrambi i livelli 0 e 1. Il pin CLK della smartcard può essere pilotato direttamente dal microcontrollore (smartcard a memoria)

oppure direttamente da un oscillatore (smartcard a microprocessore). Per questo motivo, le porte NAND IC4B, IC4C e IC4D sono collegate in modo da formare un semplice multiplexer a due linee di segnale e due linee di controllo.
Mediante questo multiplexer è possibile scegliere il segnale di clock da fornire alla smartcard tra quello proveniente dall'oscillatore (IC4A+Q1) a 3.579MHz e quello proveniente direttamente dal microcontrollore (SC_SCLK) e precisamente dal pin RA0.
Questa selezione viene effettuata dalle linee RA6 e RA7: se RA6=1 e RA7=0, l'uscita della porta NAND IC4D è sempre pari ad 1, mentre l'uscita della porta IC4B è il negato del segnale di clock proveniente dall'oscillatore IC4A+Q1. Quindi, all'uscita della porta IC4C, cioé all'ingresso del pin CLK della smartcard, ci sarà un clock a 3.579MHz, utilizzato dalle smartcard a microprocessore. Viceversa, nel caso RA6=0 e RA7=1, il segnale all'uscita della porta IC4C, quindi all'ingresso del pin CLK della smartcard, sarà uguale al segnale SC_SCLK proveniente direttamente dal microcontrollore ed utilizzato nelle smartcard a memoria. L'oscillatore basato sul quarzo Q1 e il multiplexer sono realizzati mediante quattro porte NAND, tutte presenti nell'integrato 74HC00 (IC4).
Il led rosso DL2 è pilotato direttamente dal 16F628 e può essere usato per scopi generici che possono variare a seconda dell'applicazione. Nel prossimo articolo, rappresenterà un utilizzatore, per esempio, un apricancello comandato dal lettore di smartcard in funzione della carta inserita.

REALIZZAZIONE

In figura 3 è mostrato il lato rame del nostro prototipo, in scala 1:1, sbrogliato su singola faccia. Esso può essere usato per la realizzazione del dispositivo mediante fotoincisione o tecnica analoga. La dimensione del cir-

cuito è di 10x8 cm, mezza eurocard (10x16 cm).
Dopo aver realizzato il circuito stampato, procedete con la foratura dei vari pin di tutti i componenti. I fori più grandi sono necessari per alloggiare il connettore della seriale (J2) e della smartcard (J3). Se volete, potete anche fare quattro fori ai quattro angoli per eventuale fissaggio ad un contenitore o per montare dei distanziatori. Tenete presente, però, che questi fori non sono presenti in Figura 3.
Aiutandovi con lo schema di montaggio in figura 4 e con l'elenco componenti, procedete con la saldatura di tutti i componenti.

Figura 3: *Lato rame in scala 1:1*

Figura 4: *Schema di montaggio*

Per comodità, potete iniziare con la saldatura dei componenti a basso profilo, come i tre ponticelli, le resistenze e i diodi. Proseguite con gli zoccoli per gli integrati IC2, IC3 ed IC4, con i condensatori, stando ben attenti alle polarità di quelli elettrolitici, con i led e i connettori. Il regolatore di tensione 7805 (IC1) può essere montato tranquillamente "in piedi" senza alcuna aletta di raffreddamento, poiché assorbe una potenza minima e non scalda in modo pericoloso. È importante migliorare lo smaltimento di calore di questo integrato se utilizzate tensioni d'ingresso molto grandi oppure se modificate il circuito in modo che assorba correnti elevate.
In figura 5 potete vedere una foto del prototipo dell'UniReader con una scheda Agip bianca inserita nello slot ISO7816,, il cavo di collegamento per il PC ed il cavo di alimentazione.

SVILUPPO DEL FIRMWARE

L'utilizzo di un dispositivo programmabile come il microcontrollore PIC16F628, prevede la scrittura di un firmware per tale componente.
Nel caso dell'UniReader, è necessario sviluppare un firmware che gestisca la eventuale connessione seriale RS232, il multiplexer per la scelta del segnale di clock da inviare alla smartcard, il protocollo di comunicazione con la carta eventualmente inse-

rita. Il nostro prototipo prevede anche la gestione di un led che può rappresentare, in un'applicazione pratica, un relé o qualche altro circuito di controllo.
Il microcontrollore PIC16F628 può essere programmato direttamente in assembler, poiché ha un set di istruzioni compatto e semplice da capire e imparare. Tuttavia, per il prototipo dell'UniReader, ho scelto di utilizzare il linguaggio C poiché più intuitivo, leggibile e portabile. È noto che un programma scritto con un linguaggio ad alto livello, appunto come il C, occupa sicuramente più spazio rispetto allo stesso programma scritto direttamente in assembler. Nel nostro caso, i compiti del microcontrollore sono molto semplici e quindi la ridondanza aggiunta dal compilatore non genera mai un codice troppo grande rispetto alla memoria programma del 16F628. Se volete complicare il firmware, aggiungendo altre gestioni, probabilmente sarà necessario prendere in considerazione la possibilità di scrivere il programma direttamente in assembler per evitare eseguibili troppo grandi, oppure l'utilizzo di un dispositivo con una maggiore memoria.
Come compilatore C ho scelto il CC5X (www.bknd.com) che ha una edizione FREE, in cui sono assenti alcune ottimizzazioni ed utilizzabile solo in applicazioni non commerciali, restrizioni che non creano alcun problema nella nostra situazione. Per maggiori informazioni su questo compilatore, leggete il riquadro apposito in questo stesso articolo.
Il firmware utilizzato per il test di UniReader e quelli che verranno presentati negli articoli successivi possono essere tutti scaricati dal sito di Fare Elettronica. Essi sono presenti sia in formato sorgente (file con estensione .C), sia in formato eseguibile già compilati (file con estensione .HEX). Tutti coloro che non fossero interessati allo sviluppo del firmware possono utilizzare semplicemente i file eseguibili per programmare il microcon-

Elenco componenti

Sigla	Valore
R1, R2	330 Ω 1/4 W
R3, R4	10 KΩ 1/4 W
R5	1 MΩ 1/4 W
C1, C4, C5, C11÷C13	100 nF ceramico
C2, C3	10 µF elettrolitico
C6÷C10	1µF elettrolitico
C14, C15	22 pF ceramico
D1	1N4004 (o equivalente)
DL1	Led verde 5 mm
DL2	Led rosso 5 mm
IC1	7805
IC2	MAX232 + zoccolo 16 pin
IC3	PIC 16F628 + zoccolo 18 pin
IC4	74HC00 + zoccolo 14 pin
J1	Morsettiera a due vie
J2	Connettore D-SUB9 femmina 90° per cs
J3	Connettore ISO7816
Q1	Quarzo 3.579 MHz
S1	Deviatore

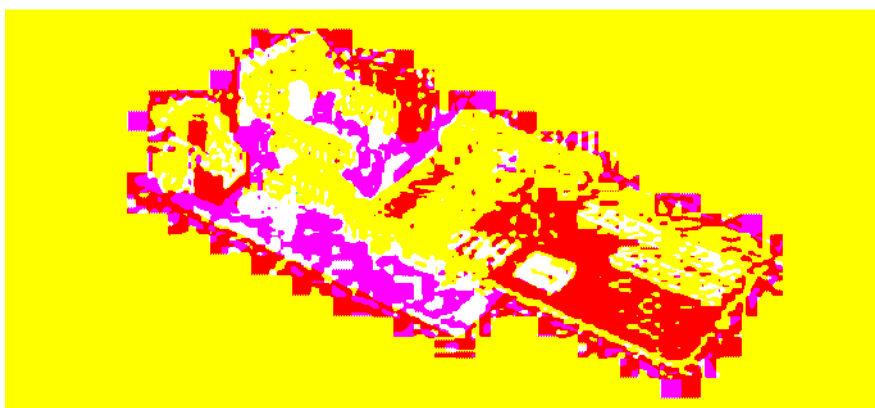



Figura 5: *Foto del prototipo di UniReader*

trollore ed utilizzarlo subito nel circuito. Se siete interessati a capire il funzionamento del firmware, oppure a modificarlo secondo le vostre necessità, potete consultare e modificare i file sorgenti relativi.

COLLAUDO

Prima di dare alimentazione al circuito, controllate eventuali cortocircuiti fra piste, soprattutto tra i due contatti dell'alimentazione del connettore J1. Controllate anche che non ci siano cortocircuiti tra l'uscita del 7805 (pin 3) e la massa. Per maggiore sicurezza, senza montare i tre integrati IC2, IC3, IC4 nei rispettivi zoccoli, collegate l'alimentazione e accendete il circuito mediante il deviatore S1 (se utilizzate uno "a levetta", spostate la levetta verso l'interno del circuito per accendere). Il led verde DL1 dovrebbe subito accendersi, indicando una tensione di circa 5V all'uscita del 7805; misurate tale tensione con un voltmetro. Controllate che tale tensione sia presente all'integrato IC2 (pin 16), IC3 (pin 14), IC4 (pin 14) e al connettore ISO7816. Potete anche controllare che sia presente una tensione di circa –10V sul pin 7 del MAX232 (IC2) che rappresenta il segnale di trasmissione su RS232.

TEST

Se tutto è corretto, inserite il MAX232 e il 74HC00 nei rispettivi zoccoli, programmate il microcontrollore con il firmware UNIREADER_TEST.HEX ed inseritelo nello zoccolo.
Collegate il lettore alla porta RS232 del PC mediante un cavo "pin to pin" maschio-femmina ed eseguite su PC il software *Hyperterminal,* normalmente in *Start > Programmi > Accessori > Comunicazioni > Hyperterminal > Hypertrm.exe* (se non doveste averlo, potete installarlo in un secondo momento dal CD di Windows). Nel seguito faremo riferimento a questo programma, ma tenete presente che potete utilizzare un qualsiasi altro software di emulazione terminale.
Dopo aver eseguito Hyperterminal, scegliete un nome qualsiasi da dare alla connessione, per esempio "Test". Cliccate su OK e scegliete, dal campo "Connetti", la porta seriale che avete utilizzato per il collegamento con UniReader (per esempio, "direttamente a COM2" se state utilizzando la seconda porta seriale). Cliccate su OK ed impostate i parametri della connessione, come in figura 6. Si tratta di un protocollo seriale alla velocità di 19200bps, 8 bit di dati, nessuna parità, 1 bit di stop senza controllo di flusso (naturalmente il firmware è stato scritto per utilizzare la porta USART con queste impostazioni). Cliccate su *File > Proprietà > Impostazioni > Impostazioni ASCII* ed abilitate la casella di controllo "Aggiungi avanzamento riga ad ogni ritorno a capo ricevuto".
Confermate la modifica.

A questo punto potete inserire una smartcard a memoria SLE4442 nel lettore ed accenderlo. Se tutto è corretto, dovreste visualizzare nella finestra di Hyperterminal un messaggio del tipo mostrato in figura 7.
Alla pressione del tasto 'R', inviato al lettore mediante RS232, UniReader genera un evento di reset e legge la risposta della carta al reset (ATR) che viene trasmesso sempre su RS232.

STANCHI DI GIOCARE?

Il circuito che ho proposto è solo un prototipo di lettore che potete utilizzare per divertirvi con le smartcard. Le modifiche che si possono fare sono numerosissime.
Per esempio, potete modificare il circuito in modo da utilizzare un PIC 16F84. In questo caso dovete gestire il pin MCLR come reset e aggiungere un oscillatore per il clock. Potete anche utilizzare il segnale a 3.579MHz già presente nel circuito come clock del microcontrollore.
È possibile aggiungere un relé, con relativo circuito di pilotaggio, al posto del led rosso di test, o insieme ad esso.
Potete sostituire la connessione RS232 con un collegamento USB oppure utilizzare una tastiera ed un display per l'interfaccia con l'utente.
Eseguito con successo il collaudo del lettore di smartcard, potete divertirvi a scrivere una bozza di programma per la gestione di una smartcard SLE4442: lettura e scrittura delle diverse zone di memoria e confronto del PSC.
Chi ha letto l'articolo su queste smartcard dovrebbe essere in grado di sviluppare un firmware di questo tipo senza grossi problemi. Se non avete a disposizione una SLE4442 vergine, potete utilizzare una tessera telefonica digitale oppure la

tessera punti dell'AGIP o di qualche altro distributore di benzina.
Se non ci riuscite, dovete solo aspettare il prossimo articolo…

Cosa ci aspetta nel prossimo articolo
Nei prossimi due articoli utilizzeremo l'UniReader rispettivamente in applicazioni con smartcard a memoria e smartcard a microprocessore. In entrambi i casi, sarà necessario sviluppare un firmware ad-hoc che gestisca la relativa tipologia di smartcard. Infine, faremo degli esempi concreti di utilizzo di smartcard a memoria (raccolta punti di un negozio e utilizzo docce a pagamento in un campeggio) e smartcard a microprocessore (gestione della rubrica e degli SMS di una SIM GSM).

BIBLIOGRAFIA E LINK
Compilatore C per PIC CC5X (in inglese): www.bknd.com
Maxim (in inglese): www.maxim-ic.com
Amp (in inglese): www.amp.com
Microchip (in inglese): www.microchip.com

Figura 6: *Impostazioni della connessione seriale in Hyperterminal*

Figura 7: *Messaggio trasmesso dal lettore all'accensione*

IL COMPILATORE
Il compilatore CC5X è distribuito dalla B Knudsen Data (http://www.bknd.com/) e supporta molti microcontrollori PIC della Microchip, in particolare quelli con un core a 12 e 14 bit, tra cui il PIC 16F628 utilizzato nell'UniReader. È possibile scaricare liberamente una versione gratuita del compilatore, alla pagina http://www.bknd.com/cc5x/download.shtml, dove è presente anche un manuale in PDF. Per studenti o progetti non commerciali, la versione FREE del compilatore gestisce al massimo 1024 istruzioni per modulo ed è privo della completa ottimizzazione. Potete scaricare sia una versione a 16 bit per DOS, sia una versione a 32 bit per Windows. Consiglio di scaricare quest'ultima versione (CC5XFREE.ZIP) che potrà essere utilizzata anche in una finestra DOS di Windows.
Una caratteristica molto interessante di questo compilatore è la possibilità di integrarlo completamente nel sistema di sviluppo MPLAB della Microchip, con la possibilità di scrivere il programma, compilarlo e farne il debug senza utilizzare le finestre DOS. Comunque, per brevità, nel seguito verrà illustrato il procedimento di compilazione a riga di comando.
Il pacchetto scaricato (CC5XFREE.ZIP), relativo alla versione 3.2, contiene, tra l'altro, l'eseguibile principale del compilatore (CC5X.EXE) e il file include 16F628.H (oltre a quelli relativi agli altri PIC supportati) per poter compilare un sorgente per il microcontrollore PIC 16F628. Potete decomprimere il file scaricato CC5XFREE.ZIP, per esempio nella cartella C:\CC5X ed aggiungere questo percorso al PATH mediante il comando DOS SET PATH=%PATH%;C:\CC5X.

Consiglio di creare un'apposita cartella dove conservare tutti i file relativi al firmware, come C:\UNIREADER nel caso del programma per il nostro lettore. Inoltre, copiate il file 16F628.H dalla cartella C:\CC5X in C:\UNIREADER.
Per scrivere il sorgente in C si può utilizzare un qualsiasi editor di testi, come il Blocco Note di Windows. È bene iniziare con la direttiva #pragma chip PIC16F628 per indicare al compilatore il microcontrollore per cui deve generare il file eseguibile.
Per esempio, potete scrivere il seguente programma che fa lampeggiare il led rosso dell'UniReader.

```
#pragma chip PIC16F628
/* Bit di configurazione. */
#define CP_OFF |= 0x3C00   /* Code Protection Program Memory OFF. */
#define CPD_OFF |= 0x0300  /* Data Code Protection OFF. */
#define LVP_OFF &= 0x3F7F  /* Low Voltage Programming OFF. */
#define BOD_ON |= 0x0040   /* Brown Out Detect Disabled. */
#define MCLR_OFF &= 0x3FDF /* MCLR disabled (internally tied to Vcc). */
#define PWRT_ON &= 0x3FF7  /* Power Up Timer enabled. */
#define WDT_OFF &= 0x3FFB  /* Watchdog Timer disabled. */
#define FOSC_INTRCIO |= 0x0010, &= 0x3FFC      /* Int RC without CLKOUT. */
#pragma config CP_OFF, CPD_OFF, LVP_OFF, BOD_ON, MCLR_OFF, PWRT_ON, WDT_OFF, FOSC_INTRCIO

bit LED  @ PORTB.3;   /* LED collegato al PIC. */

void
main( void )
{
  /* Metti in output il pin del LED spegnendolo. */
  PORTB  = 0xFF;
  TRISB &= 0xF7;

  while(1) {
    unsigned char i1, i2;

    /* Pausa */
    for( i1=1; i1<100; i1++ )
      for( i2=1; i2<0xFF; i2++ )
        ;
    /* Cambia lo stato del LED */
    if (LED)
      LED = 0;
   else
      LED = 1;
  }
}
```

Salvate il file come C:\UNIREADER\UNIREADER.C e compilatelo in una finestra DOS mediante il comando **CC5X UNIREADER.C** eseguito, naturalmente, all'interno della cartella C:\UNIREADER.
Se la compilazione è andata a buon fine, potete osservare un messaggio simile a quello in Figura 8. Tra le altre cose, il compilatore fornisce informazioni sull'utilizzo della RAM, sull'esito dell'ottimizzazione, sul numero di istruzioni generate, ecc. Il file eseguibile ottenuto si chiama UNIREADER.HEX e potrà essere utilizzato per pro-

grammare fisicamente il microcontrollore.

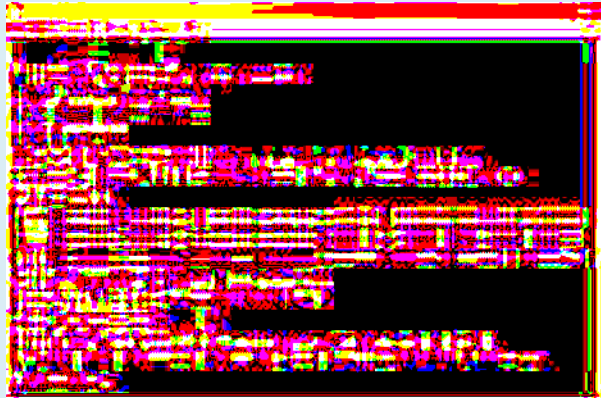



Figura 8: *Risultato della compilazione in una finestra DOS*

All'interno di questo file è inserita anche la parola di configurazione per il PIC, molto importante durante la fase di programmazione del dispositivo. Se il software che utilizzate per questo scopo non gestisce la parola di configurazione memorizzata nel file .HEX, dovete specificarla manualmente, disabilitando qualsiasi protezione (sul codice e sui dati), la programmazione a bassa tensione, l'MCLR, il watchdog.

Abilitate soltanto il Brown-Out-Detect e il Power-Up-Timer. Inoltre, selezionate come sorgente di clock l'oscillatore interno a 4 MHz senza utilizzare il pin CLKOUT come uscita di questo clock. In Figura 9 è presente la schermata di programmazione, con i bit di configurazione, del diffusissimo software IcProg.

Figura 9: *Schermata del software IcProg che mette in evidenza i bit di configurazione utilizzati per la programmazione del PIC 16F628*

HARDWARE PER LO SVILUPPO DI CPLD XILINX

terza parte

di Agostino Rolando
a.rolando@farelettronica.com

Si conclude qui la trattazione del progetto rivolto allo studio e alla sperimentazione dei linguaggi di descrizione dell' hardware, con un nuovo circuito e due esempi di codice: il primo in linguaggio Verilog, il secondo in VHDL.

In questa puntata prendiamo in considerazione un modulo contenente un convertitore digitale-analogico a 8 bit di tipo flash, da collegare al modulo-base presentato nel primo articolo della serie.

Per lo sviluppo del firmware sono stati utilizzati il tool di sintesi *XST* (Xilinx Synthesis Technology) e il tool di simulazione *Modelsim*, facenti parte del *Xilinx ISE Webpack*, versione 4.2WP3x. Entrambi sono scaricabili gratuitamente dal sito www.xilinx.com, previa registrazione.
Per il disegno dello schema e per la generazione del layout del circuito stampato si sono utilizzati gli applicativi SPLAN e SPRINT, il cui reader gratuito si può scaricare dal sito di Fare Elettronica.

MODULO DI CONVERSIONE D/A

Questo modulo va collegato al connettore di I/O del modulo base (J04).
Con esso è possibile generare differenti forme d'onda, come da esempi riportati successivamente.
Per questa funzione si è utilizzato l'integrato Intersil CA3338 in quanto impiegabile a singola alimentazione (+5V) e con la possibilità di funzionare a conversione continua, cioè senza la necessità di particolari temporizzazioni o segnali di handshake.
Lo schema elettrico è riportato in figura 1, il master del circuito stampato è in figura 2

Infine, per poter rendere udibile il segnale generato, è stato collegato un trasduttore piezoelettrico ad alta impedenza (PZ1) direttamente all'uscita analogica del convertitore.

ESEMPIO 1 GENERATORE SINUSOIDALE

Questo esempio utilizza il modulo Convertitore D/A. Mediante una tabella predefinita di 64 valori in formato binario a 8 bit, scandita sequenzialmente, viene generata una forma d'onda sinusoidale alla frequen-

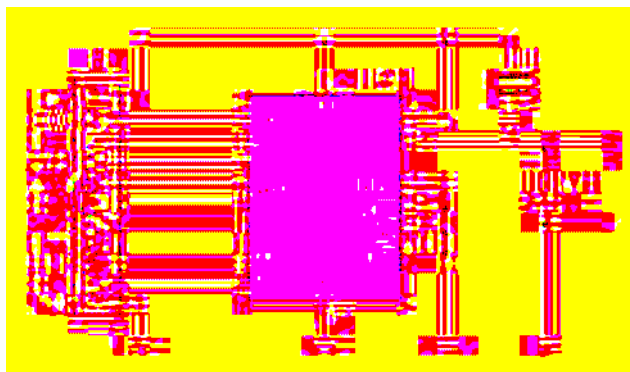



Figura 1: *Schema elettrico del convertitore*

za di 1.8 Khz.
Tali campioni sono stati preventivamente calcolati mediante un programma in linguaggio Pascal. Dal momento che il convertitore viene alimentato a singola alimentazione (+5V), nel calcolo dei campioni binari si è aggiunto un offset di 2.5 Volt. In questo modo, la sinusoide che ne risulta è traslata di questa quantità.
I campioni vengono inviati al convertitore sui pin del connettore di uscita J04, denominati D0..D7, del modulo base. Nella seguente figura 5 sono evidenziati i collegamenti tra i moduli utilizzati.

Il codice Verilog relativo all'esempio è disponibile per il download dal sito di Fare Elettronica. In figura abbiamo riportato la parte di codice che assegna i valori binari al registro di uscita "dato_out". Infine, il registro viene riportato sulle uscite D0..D7 con l'istruzione:

```
assign
{D7,D6,D5,D4,D3,D2,D1,D0}
= dato_out;
```

Il risultato dell'esecuzione di questo modulo è una sinusoide a 1.8 Khz e ampiezza Vpp=5V, con

Elenco componenti	
Sigla	**Valore**
C01	100 nF 50 V, package 0805 ceramico
IC01	Intersil CA3338, package SO-16
PZ1	Trasduttore piezoelettrico
JP1	Connettore lineare a striscia, 10 pin maschio

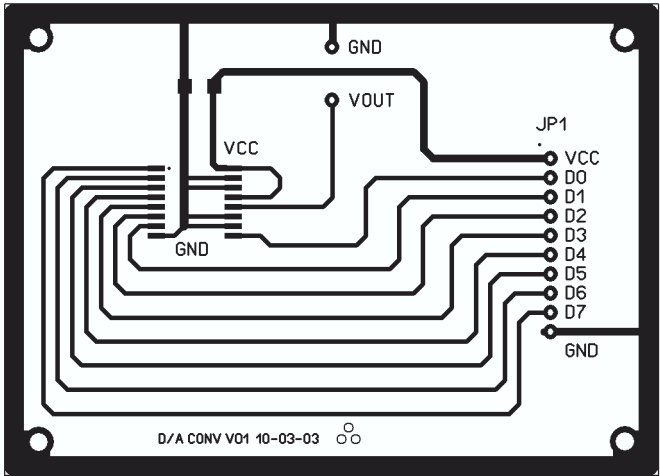



Figura 2: *Circuito stampato in scala 1:1 (lato rame) del modulo convertitore*

Figura 2a: *Prototipo del modulo convertitore (lato saldature e lato componenti)*

Figura 3: *Piano di montaggio lato saldature del modulo convertitore*

Figura 4: *Piano di montaggio lato componenti del modulo convertitore*

offset di 2,5V, come da figura 7.

Il sistema è particolarmente flessibile. Infatti, se si dispone dei campioni in formato binario, è possibile generare una forma d'onda qualsiasi (vedere figure 8, 9 e 10).

Per il calcolo dei campioni, può essere d'aiuto servirsi di un foglio Excel, come da figura 11. In questo esempio, nella colonna A sono stati inseriti i 64 campioni della sinusoide, mentre nella colonna B sono stati riportati i campioni della forma d'onda a scala.

Per ottenere i grafici a partire dai campioni, si seleziona una colonna di dati e si esegue il Chart Wizard di Excel (figura 12).

In questo modo si può creare e visualizzare la forma d'onda desiderata, costruita campione per campione.
Una volta che i valori siano stati verificati e assodati, si può intervenire sul codice Verilog del generatore di segnale, modificando opportunamente la tabella che verrà utilizzata dal convertitore D/A.

Per semplificare il procedimento, è possibile anche inserire i dati direttamente in formato decimale, anziché binario, come si è visto nell' esempio del generatore sinusoidale.
Ad esempio, per quanto riguarda la forma d'onda a scala, possiamo utilizzare il formato riportato nel tratto di codice seguente, dove 4'dxxxx indica un numero a 4 cifre in formato decimale:

Figura 5: *Collegamento tra moduli*

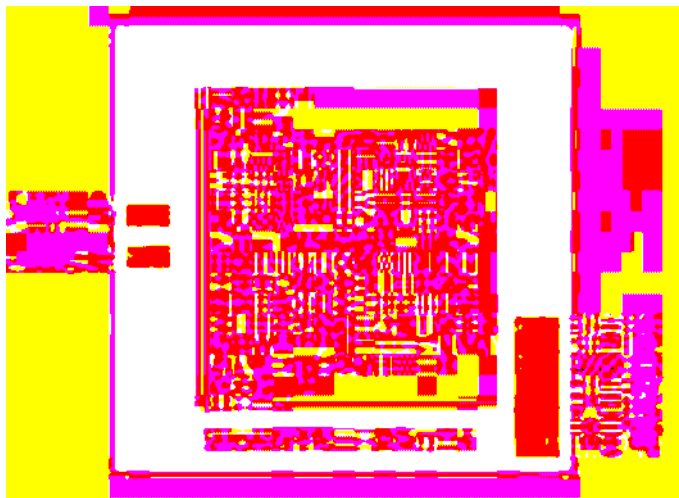



Figura 6: *Codice Verilog per il generatore sinusoidale*

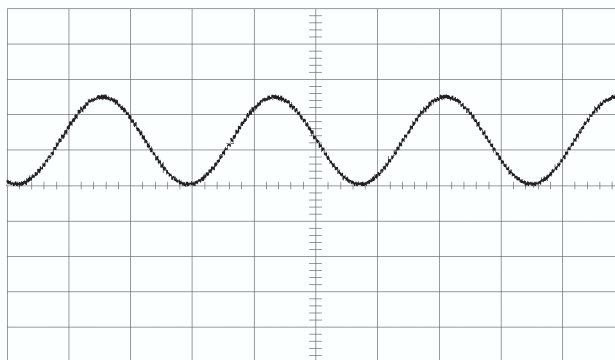



Figura 7: *Forma d' onda sinusoidale*

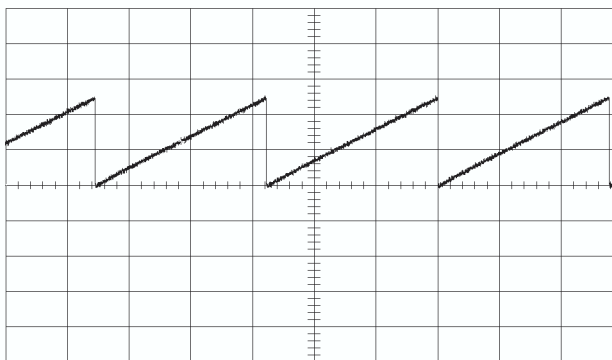



Figura 8: *Forma d' onda a dente di sega*

```
// Tabella campioni scala

always@ ( conta )
 begin
  case(conta)
   {(6'b000000)} :
dato_out <= 4'd0000;
   {(6'b000001)} :
dato_out <= 4'd0000;
   {(6'b000010)} :
dato_out <= 4'd0000;
   {(6'b000011)} :
dato_out <= 4'd0000;
   {(6'b000100)} :
dato_out <= 4'd0000;
   {(6'b000101)} :
dato_out <= 4'd0000;
   {(6'b000110)} :
dato_out <= 4'd0000;
   {(6'b000111)} :
dato_out <= 4'd0000;
   {(6'b001000)} :
dato_out <= 4'd0000;
   {(6'b001001)} :
dato_out <= 4'd0000;
   {(6'b001010)} :
dato_out <= 4'd0050;
   ...
   {(6'b111100)} :
dato_out <= 4'd0250;
   {(6'b111101)} :
dato_out <= 4'd0250;
   {(6'b111110)} :
dato_out <= 4'd0250;
   {(6'b111111)} :
dato_out <= 4'd0250;
   default : dato_out <=
4'd0000; // 0
  endcase
end  // always
```

Figura 9: *Forma d' onda quadra*

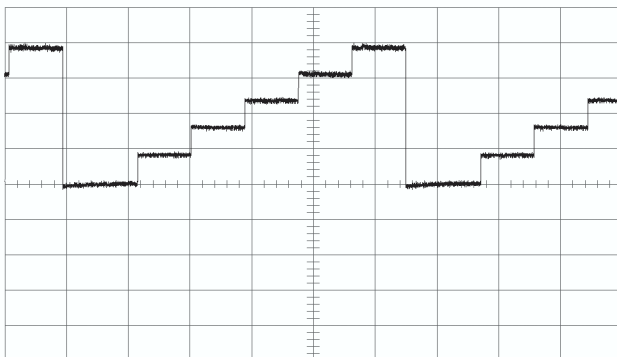



Figura 10: *Forma d' onda a scala*

Figura 11: *Foglio Excel per il tracciamento delle funzioni*

Un caso particolare è rappresentato dalla forma d'onda a dente di sega, che abbiamo visto in figura 8. Per realizzarla, è sufficiente sostituire la riga di istruzione:

```
  assign
{D7,D6,D5,D4,D3,D2,D1,D0}
= latch_dato;
```

con la seguente parte di codice:

```
assign  D7 = conta[5];
assign  D6 = conta[4];
assign  D5 = conta[3];
assign  D4 = conta[2];
assign  D3 = conta[1];
assign  D2 = conta[0];
```

```
assign  D1 = 1'b0;
assign  D0 = 1'b0;
```

il che vuol dire inviare, in ingresso al convertitore D/A, non i campioni sinusoidali ma le uscite del contatore che scandisce la tabella dei campioni. I bit meno significativi, D1 e D0,sono fissati a zero.
Per generare l'onda quadra, si scriverà invece:

```
assign  D7 =
latch_dato[7];
assign  D6 =
latch_dato[7];
assign  D5 =
latch_dato[7];
assign  D4 =
latch_dato[7];
assign  D3 = 1'b0;
assign  D2 = 1'b0;
assign  D1 = 1'b0;
assign  D0 = 1'b0;
```

Questo pacchetto di assegnazioni effettua, in pratica, un filtro a soglia, trasformando quella che in origine è una sinusoide in un'onda quadra della stessa frequenza.

ESEMPIO 2 UART

Questo esempio è stato scritto in linguaggio VHDL e realizza la parte di ricezione di un dispositivo **UART**.
Si prevede di collegare il modulo base, attraverso la propria interfaccia rs232, a un programma di emulazione di terminale VT100, per esempio dal proprio PC. A questo punto, alla pressione di tasti sulla tastiera del PC corrisponde l'invio del codice ASCII del codice corrispondente al carattere battuto e questo codice viene reso disponibile sui pin di uscita D0÷D7 del modulo base stesso.
Per la visualizzazione della codifica ASCII è conveniente utilizzare il modulo a Barra di led già descritto in una delle puntate precedenti.

In questo modo, il modulo base può fungere da attuatore remoto di comandi inviati su linea seriale.
Un utilizzo pratico può consistere, ad esempio, nel comandare luci o relé a distanza.

I parametri di configurazione della linea seriale sono prefissati a: Velocità = 9600 baud, Bit di start = 1, Bit di stop = 1, lunghezza di parola = 8 bit, parità = nessuna.
Come già accennato, la logica di funzionamento prevede di pilotare le otto linee di output presenti sul modulo base a seguito di digitazione di caratteri ASCII da effettuarsi con la tastiera del PC.

In particolare, il codice prevede la possibilità di pilotare N pacchetti da 8 bit, poiché permette di specificare anche un indirizzo identificativo di pacchetto-dato. Il limite all'espandibilità è imposto dal numero di pin di I/O offerti dalla particolare CPLD adottata.
In definitiva, i caratteri da premere sulla tastiera sono due: il primo identifica l'indirizzo del pacchetto, il secondo identifica il dato vero e proprio, che consiste nel codice ASCII del tasto stesso. In questa occasione, si è fissato l'indirizzo al valore uno, cioè alla codifica Ascii 00110001.

Struttura del codice

Nel dettaglio, il codice ha una struttura gerarchica e si compone di tre moduli: un modulo principale (TOP), un modulo per la ricezione seriale (RX_RS232) e un modulo di decodifica dei comandi inviati da tastiera (MUX_DECODER).

Figura 12: *Impiego del Chart Wizard di Excel*

Il codice VHDL corrispondente è disponibile per il download sul sito di Fare Elettronica.
A differenza di quanto fatto in Verilog, in questo caso il pin-out viene specificato direttamente in testa al codice sorgente, con l'istruzione attribute, ad esempio:

```
  attribute pin_assign of
CLK : signal is "5";
```

Nella figura 13 vengono evidenziati i collegamenti tra i moduli hardware utilizzati, insieme al cavo seriale di interconnessione con il PC.

CONCLUSIONI

Una volta collaudati e perfezionati i singoli moduli, l'intero progetto è stato realizzato su di un'unico stampato (figure 14 e 15), in cui è stata inclusa anche l'interfaccia JTAG già presentata nel numero di febbraio 2004.
La serigrafia (lato superiore) fornisce una traccia per i collegamenti, da connettore a connettore, che si possono effettuare a partire dal modulo-base verso i vari moduli di espansione.

A seconda dell' applicazione che si desidera realizzare, si collegherà il modulo-base (al connettore J04) con l'accessorio desiderato (al rispettivo connettore JP1), per mezzo di un apposito cavetto pin-to-pin, con connettori a striscia femmina.
Analogamente, la tastierina andrà collegata al connettore J05 del modulo-base e, riguardo l'interfaccia JTAG, si disporrà un cavetto a sei vie a partire dal connettore J02 del modulo-base verso il J03 del modulo JTAG.
Inoltre, si dovrà predisporre una connessione flat femmina-maschio a 25 pin per mettere in comunicazione il connettore J01 del modulo JTAG con la porta parallela del PC, per il trasferimento del firmware di programmazione.

Figura 13: *Collegamento tra i moduli utilizzati*

Vi ricordo che i sorgenti realizzati per questo progetto è possibile scaricarli dal sito di Fare Elettronica.

Figura 14: *Prototipo lato componenti*

Figura 15: *Prototipo lato saldature*

PRATICAMENTE... IL TIMER 555: METRONOMO ELETTRONICO

di Maurizio Del Corso
m.delcorso@farelettronica.com

Introdotto nel 1971, venne chiamato "The IC Time Machine" e fu il primo timer integrato della storia. Dopo oltre 30 anni, il timer 555 viene ancora utilizzato in svariate applicazioni e per questo motivo la puntata di "Praticamente" di questo mese è interamente dedicata a questo dispositivo: la sua struttura interna, le configurazioni come astabile e monostabile e, come esempio di applicazione, la realizzazione di un metronomo elettronico.

ANALISI E SPECIFICHE DEL PROBLEMA

La definizione di metronomo secondo il dizionario della lingua italiana è la seguente:

<< s. m. strumento di misura usato spec. per battere il tempo in musica: metronomo meccanico, meccanismo a orologeria con carica a molla e scappamento a pendolo volutamente rumoroso; metronomo elettronico, dispositivo che comprende un generatore di impulsi e un piccolo altoparlante. >>

Si deve quindi realizzare un dispositivo in grado di emettere un "tic" da un altoparlante ad una cadenza ben definita e regolabile mediante un potenziometro.

LA SOLUZIONE PROPOSTA

Lo schema proposto in figura 1 soddisfa i requisiti richiesti. Analizziamone il funzionamento in modo dettagliato.

Figura 1: La soluzione proposta

Il circuito è un generatore di onda quadra (multivibratore astabile) la cui frequenza dipende dal tempo che il condensato-

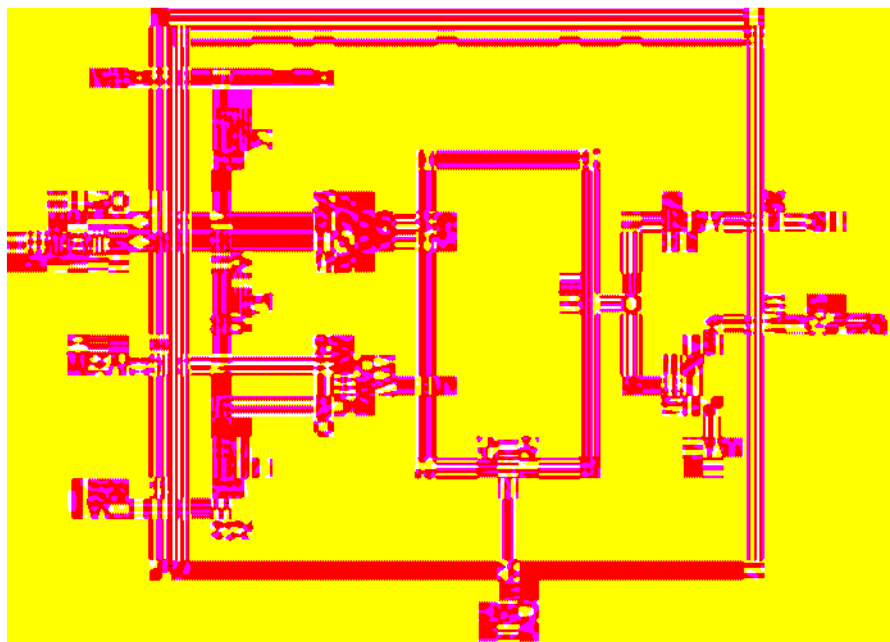



Figura 2: Struttura del timer 555

re C1 impiega per caricarsi e scaricarsi. Poiché la corrente di carica e scarica è limitata dalle resistenze P1 ed R1, regolando il potenziometro P1 si varia la frequenza dell'onda quadra, quindi del "tic" emesso dall'altoparlante. Il condensatore C2 ha il compito di bloccare la corrente continua ed impedire il danneggiamento dell'altoparlante.

DENTRO IL 555

Per comprendere a fondo il funzionamento del timer 555, è necessario conoscere la sua struttura interna che è riportata nella figura 2.
La tensione di alimentazione Vcc viene ripartita in tre parti uguali mediante tre resistenze da 5KOhm (da qui il nome 555) ed applicata ai due comparatori come tensione di riferimento. Il comparatore A ha quindi 2Vcc/3 come tensione di riferimento al piedino non invertente, mentre il comparatore B ha Vcc/3 come tensione di riferimento al terminale non invertente. L'uscita dei due comparatori pilota un flip-flop SR il cui funzionamento è il seguente: se R=1 e S=0 l'uscita /Q è alta, viceversa se R=0 e S=1 l'uscita /Q è bassa. Se R ed S sono entrambi a livello basso, l'uscita non ha alcuna variazione. R=1 ed S=1 è un caso da evitare in quanto l'uscita /Q verrebbe a trovarsi in uno stato non definito.
L'uscita /Q del flip-flop viene portata all'esterno mediante una porta NOT, ciò significa che se R=1 ed S=0 l'uscita del timer è bassa mentre se R=0 ed S=1 l'uscita del timer è alta. Si noti che l'uscita del flip-flop è collegata alla base di un transistore, quindi se /Q=0 il transistore è interdetto (quindi è come se non fosse presente nel circuito) e se /Q=1 il transistore è in saturazione per cui si comporta come un corto circuito tra collettore ed emettitore.

Timer 555 come Trigger di Schimtt

Del trigger di Schmitt abbiamo ampiamente discusso il mese scorso: esso è un comparatore con isteresi, quindi con due differenti tensioni di soglia. È possi-

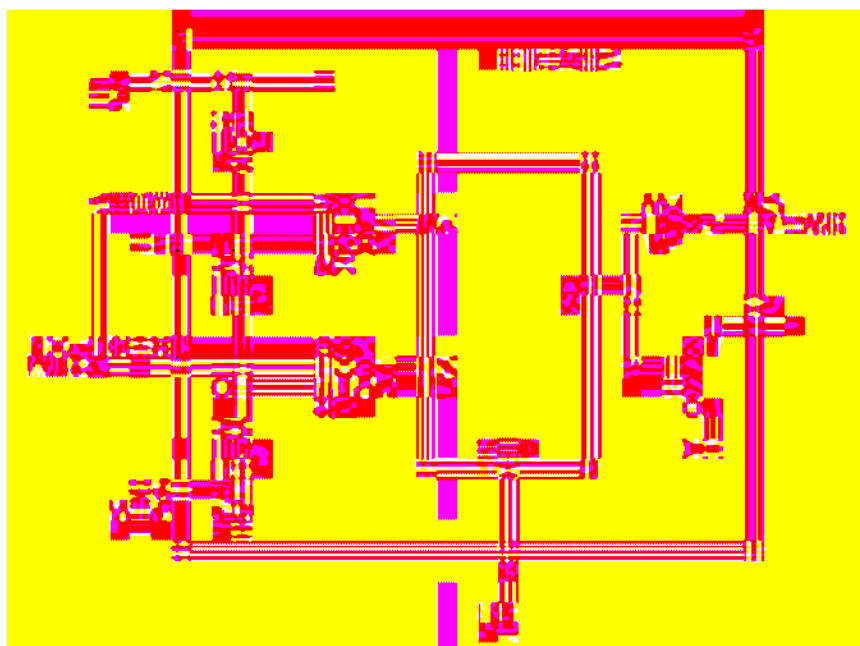



Figura 3: Trigger di Schmitt con Timer 555

bile realizzare un trigger di Schmitt utilizzando un timer 555 seguendo lo schema di figura 3. Le forme d'onda nel caso in cui venga applicato all'ingresso un segnale sinusoidale, sono riportate in figura 4: si noti che le tensioni di riferimento sono appunto Vcc/3 e 2Vcc/3. Analizziamo il funzionamento passo dopo passo con riferimento alle figure 3 e 4. I pin Trigger e Threshold sono collegati tra loro per cui la tensione di ingresso è identica per entrambi i comparatori.
Inizialmente la tensione Vin è nulla per cui avremo R=0 ed S=1, quindi l'uscita al pin 3 è alta. Quando la sinusoide in ingresso supera il valore Vcc/3 della prima tensione di soglia, il comparatore B porta a zero la sua uscita quindi il flip-flop rimane nello stato di conservazione e l'uscita del timer non cambia.
Quando Vin supera la seconda tensione di soglia 2Vcc/3, il comparatore A porta a livello alto la sua uscita quindi il flip-flop si resetta portando a zero l'uscita del timer.

Quando la Vin è in diminuzione e scende al di sotto di 2Vcc/3, il comparatore A riporta a zero la sua uscita ed il flip flop entra nello stato di conservazione mantenendo inalterata l'uscita del timer. Solo quando la sinusoide di ingresso scende al di sotto di Vcc/3 l'uscita del timer torna alta, in quanto il comparatore B commuta portando S a livello alto.

Figura 4: *Forme d'onda relative al trigger di Schmitt di figura 3 con un segnale sinusoidale in ingresso*

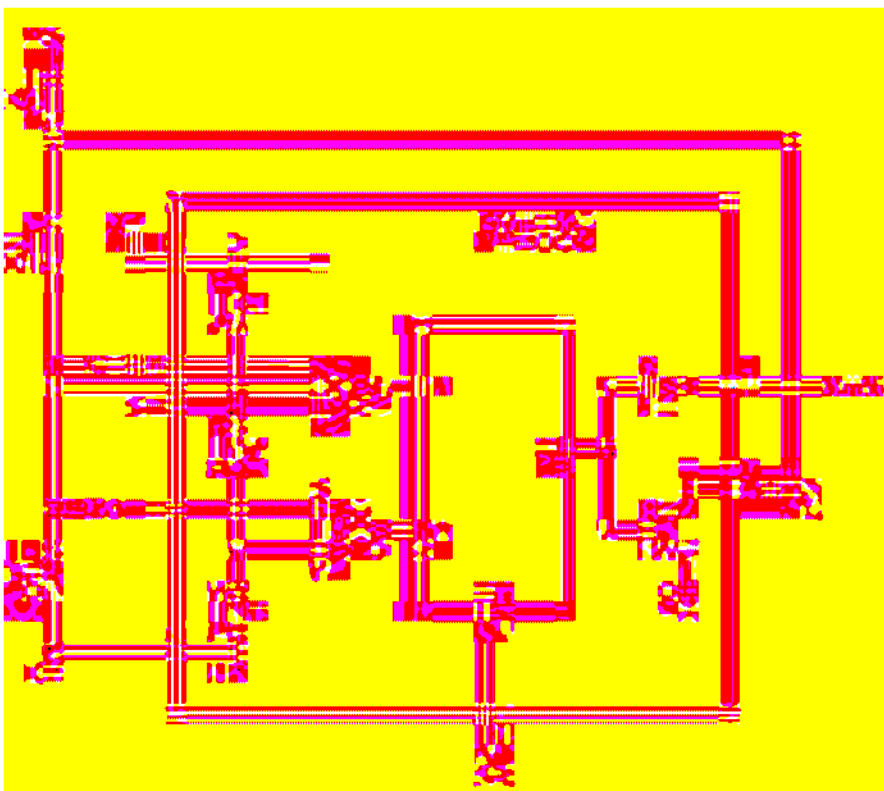



Figura 5: *Timer 555 in configurazione astabile*

Timer 555 come multivibratore astabile

Questa è l'applicazione del timer 555 che è stata usata per la realizzazione del metronomo elettronico. Lo schema è quello di figura 5 e, come si può notare, sono necessarie due resistenze ed un condensatore oltre, ovviamente, al timer 555.

Come si può intuire dal suo nome, un multivibratore astabile è un circuito che non possiede uno stato stabile per cui oscilla continuamente tra due stati un po' come due bambini su un'altalena. L'uscita di un

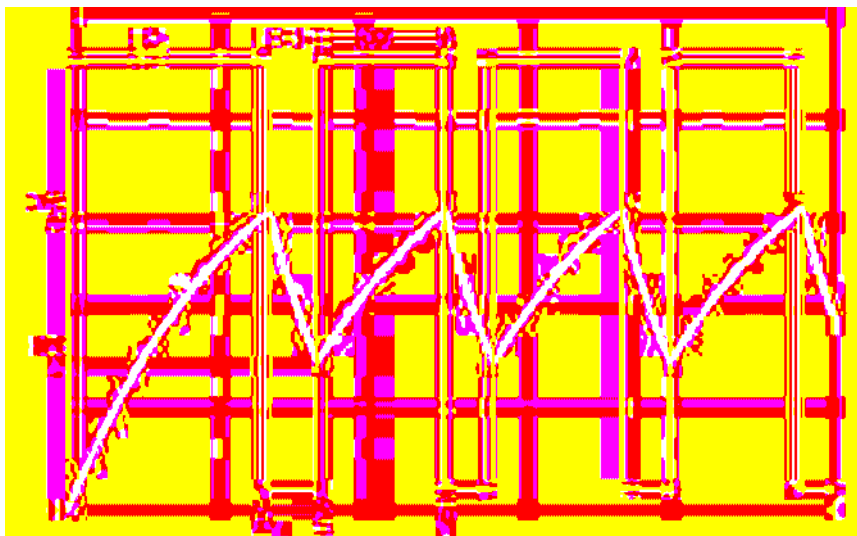



Figura 6: Forme d'onda dell'astabile di figura 5

multivibratore astabile è dunque un'onda quadra la cui frequenza dipende dalle resistenze e dal condensatore esterni al timer. Per capire in che modo viene generata l'onda quadra, si faccia riferimento allo schema di figura 5 ed alle relative forme d'onda riportate in figura 6 in cui in blu è riportata l'uscita del timer ed in rosso la tensione ai capi del condensatore (applicata anche all'ingresso dei due comparatori).

Supponiamo che il condensatore sia inizialmente scarico (Vc=0). Se Vc=0, per effetto dei due comparatori si ha R=0 e S=1 quindi Vout è a livello alto ed il transistor interno al timer è interdetto (perché /Q=0). L'interdizione del transistor e la corrente nulla assorbita dagli ingressi degli operazionali, fanno sì che il condensatore C sia libero di caricarsi attraverso

le resistenze Ra ed Rb. La tensione Vc inizia quindi ad aumentare e, quando supera il valore di soglia Vcc/3 il comparatore B commuta portando S a zero quindi il flip flop entra in conservazione mantenendo inalterato lo stato dell'uscita. Quando invece Vc supera il valore 2Vcc/3 (istante t_A in figura 6), il comparatore A porta R a livello alto e l'uscita Vout commuta a valore basso (quindi /Q a valore alto).
Questo provoca la saturazione del transistore di scarica che porta il pin 7 a massa innescando la scarica del condensatore C attraverso Rb, quindi la conseguente diminuzione della tensione Vc che riporta il flip flop nello stato di conservazione. Quando Vc scende al di sotto di Vcc/3 (istante t_B in figura 6) si ha una nuova commutazione di Vout a livello alto che interdice il transistore e permette al condensatore di iniziare nuovamente a caricarsi, avviando un nuovo ciclo. A parte il transitorio iniziale, il condensatore impiegherà sempre il tempo T1 per scaricarsi ed il tempo T2 per caricarsi. La somma T=T1+T2 è il periodo dell'onda quadra, quindi f=1/T la sua frequenza. I tempi T1 e T2 sono dati rispettivamente da:

$$T1=0{,}693\cdot Rb\cdot C$$
$$T2=0{,}693\cdot (Ra+Rb)\cdot C$$

Si noti che mentre il tempo di carica dipende da Ra ed Rb, quello di scarica dipende esclusivamente da Rb. Dalle due espressioni di T1 e T2 possiamo ricavare l'espressione della frequenza dell'onda quadra che è data da:

$$f=1/(T1+T2)=1{,}44/(Ra+2Rb)C$$

Figura 7: Calcolo delle resistenze e del condensatore per una determinata frequenza

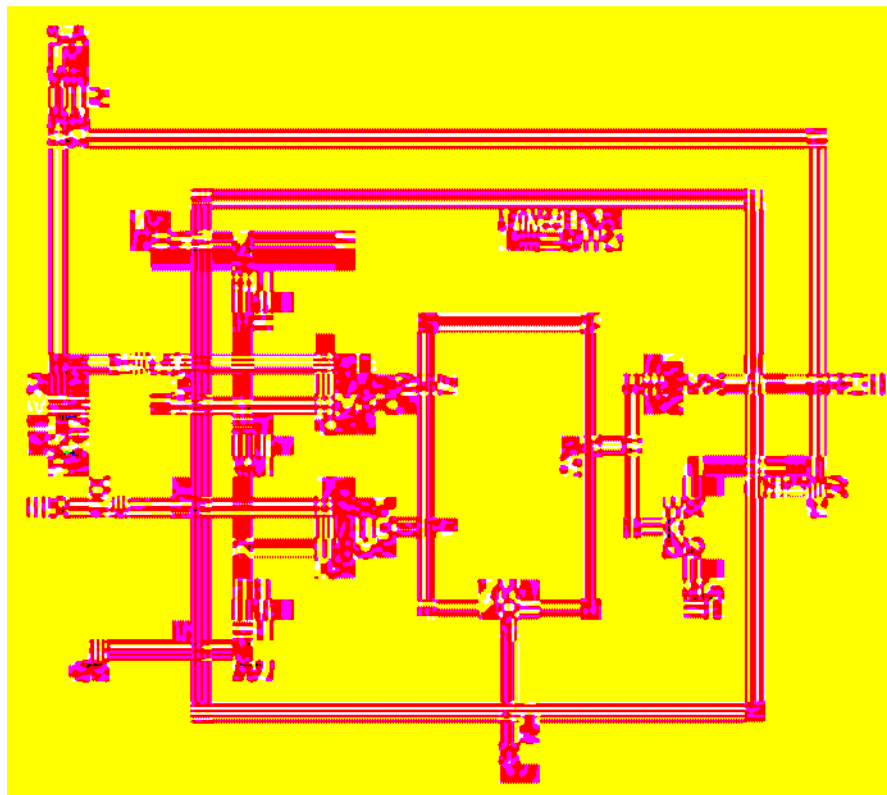



Figura 8: Timer 555 in configurazione monostabile

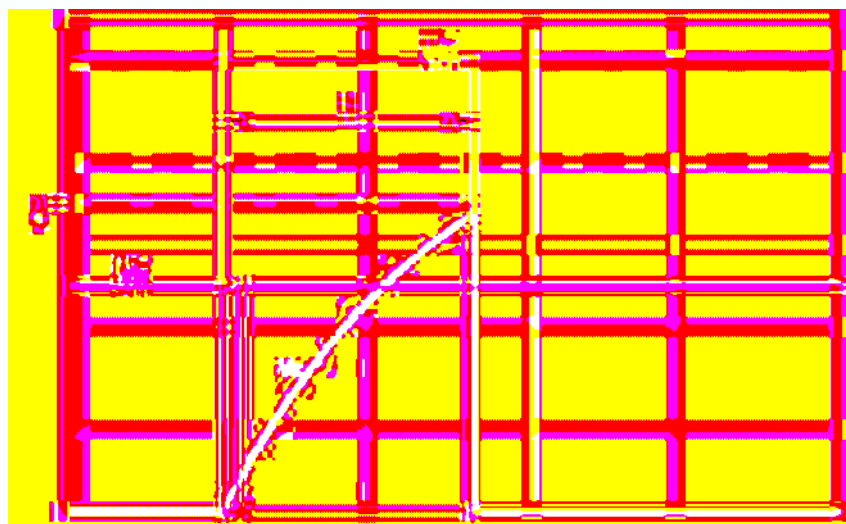



Figura 9: Forme d'onda del monostabile di figura 8

Per determinare i valori delle resistenze e dei condensatori da inserire nel circuito per ottenere una determinata frequenza, esistono dei grafici come quello di figura 7. Supponiamo di volere una frequenza di 100Hz.
Seguendo la linea verticale a 100Hz si va ad incontrare una delle linee oblique che rappresentano il valore della somma Ra+2Rb. Una volta scelto il valore della somma dei resistori, dobbiamo spostarci orizzontalmente fino a leggere il valore del condensatore sull'asse verticale del grafico. Dunque per i 100Hz, scegliendo 1MOhm come somma di Ra+2Rb si ottiene per il condensatore un valore di poco superiore agli 0.01μF.

È importante notare che comunque si scelgano i componenti esterni, non sarà possibile ottenere T1=T2 ovvero un'onda quadra con duty cycle del 50% (il duty cycle di un'onda quadra è dato dal rapporto percentuale tra il tempo in cui l'onda è a livello alto ed il periodo totale: $d=100\cdot T1/(T1+T2)$).

Timer 555 come multivibratore monostabile

Un multivibratore monostabile riceve in ingresso un impulso di "start" (detto segnale di trigger) ed in uscita genera un impulso di durata prefissata. Per usare un timer 555 in configurazione monostabile, si deve usare il circuito di figura 8 in cui R e C determinano la durata dell'impulso in uscita. L'impulso di trigger viene applicato al pin 2 e deve essere una transazione alto-basso. In figura 9 le forme d'onda relative al multivibratore monostabile.
Con Vin=0 e /Q=1 siamo nello stato di riposo in quanto S=0, R=0 ed il condensatore è mantenuto scarico in quanto il transistor è in saturazione. In questo stato l'uscita del timer è Vout=0. Se Vin ha una commutazione verso il livello basso, anche per un tempo minimo (forma d'onda in verde in figura 9), il comparatore B porta a livello alto l'ingresso S e l'uscita commuta a livello alto provocando contemporaneamente l'interdizione del transistor di scarica. In questo modo il condensatore è libero di caricarsi attraverso la resistenza R. La tensione di uscita rimane alta fintanto che la tensione sul condensatore (in rosso in figura 9) non supera i 2Vcc/3: in questo istante il flip-flop viene resettato e l'uscita torna a livello basso. Il transistor entra in conduzione scaricando velocemente il condensatore che rimane bloccato ad una tensione prossi-

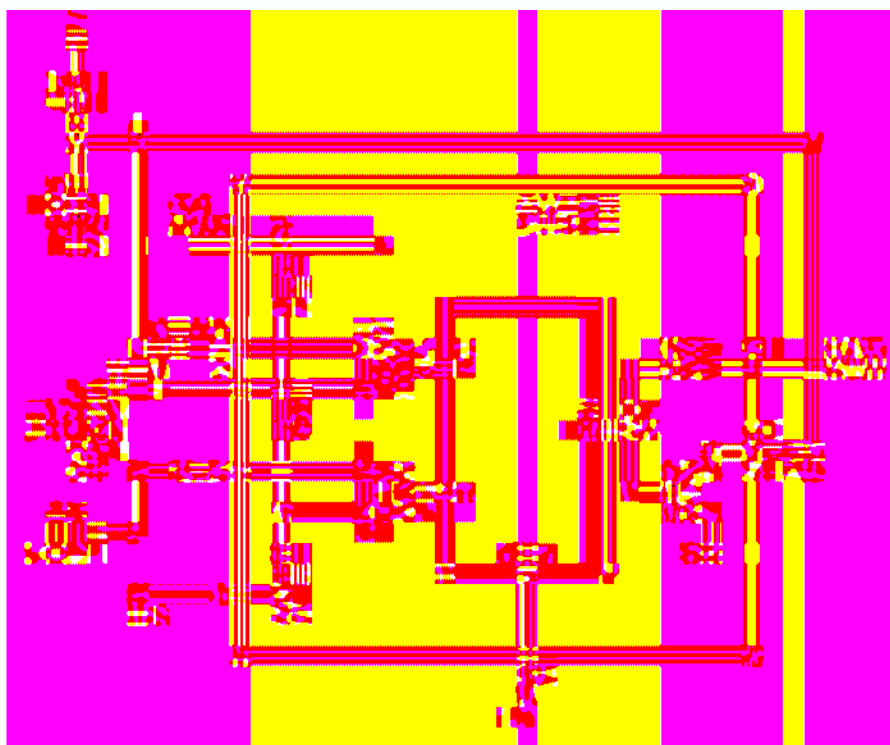



Figura 10: *Modulatore PWM con timer 555: Vmod è la tensione modulante*

Figura 11: *Forme d'onda relative al modulatore PWM con timer 555*

ma allo zero e qui vi rimane fino ad un nuovo impulso sull'ingresso. La durata T dell'impulso in uscita dipende dal condensatore e dalla resistenza, secondo la formula

$$T=1{,}1 \cdot R \cdot C$$

Modulazione PWM

PWM è l'acronimo di Pulse Width Modulation ovvero la generazione di un treno di impulsi in cui la durata di ciascun impulso dipende dal valore della tensione di ingresso Vmod (tensione modulante). Con il timer 555 è possibile ricavare un modulatore PWM utilizzando la configurazione monostabile, pilotandola in continuazione con impulsi di trigger ed applicando la Vmod al pin 5 come mostrato in figura 10. La Vmod va a sommarsi alle tensioni di riferimento variando così la durata dell'impulso. La forma d'onda così ottenuta è quella in blu di figura 11.

IL QUESITO DEL MESE

Come quesito del mese propongo il multivibratore astabile modificato riportato in figura 12. Sapreste dire in che modo la modifica apportata influisce sul funzionamento del circuito?

Occupiamoci ora della soluzione del quesito del mese scorso: l'interruttore crepuscolare.

Come promesso pubblichiamo la soluzione sviluppata da un lettore e ritenuta la più interessante, ovvero quella di Gianni Pasqualetti di Firenze. Lo schema proposto da Gianni è riportato in figura 13 ed effettivamente è la soluzione più semplice al problema proposto. La fotoresistenza è inserita in un partitore resistivo quindi la tensione all'ingresso del comparatore vale $Vcc \cdot R/(R+R1)$. In caso di buio è R=1KΩ per cui scegliendo R1=1KΩ la tensione in ingresso al comparatore vale Vcc/2. Avendo scelto questo valore per R1, in caso di piena luce (R=100Ω) la tensione vale $0.09 \cdot Vcc$.

Scegliendo Vcc=12V la tensione di ingresso del comparatore varia da 6V in caso di buio fino a 1.08V in caso di luce. Il partitore costituito da R2 e R3 fornisce la tensione di riferimento per il comparatore che può essere variata agendo sul trimmer R3. Con i valori riportati in figura 13, tale tensione può essere variata da 0 fino a 2Vcc/3 (quindi 8V). Questa tensione costituisce la soglia di eccitazione del relé quindi permette di regolare a quale livello di buio eccitare il relé. Il relé viene pilotato tramite il

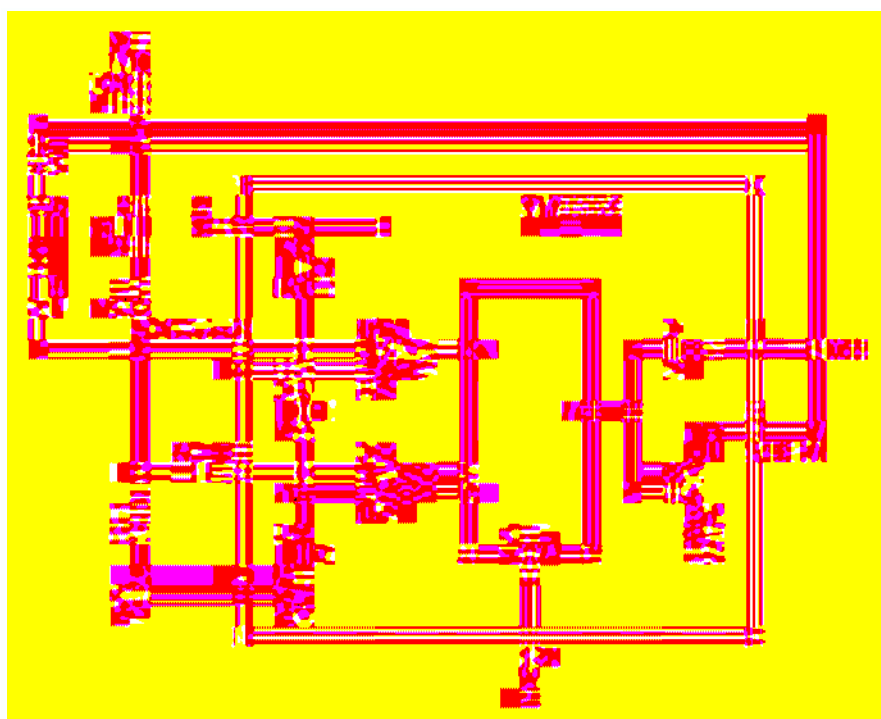



Figura 12: L'astabile modificato per il quesito del mese

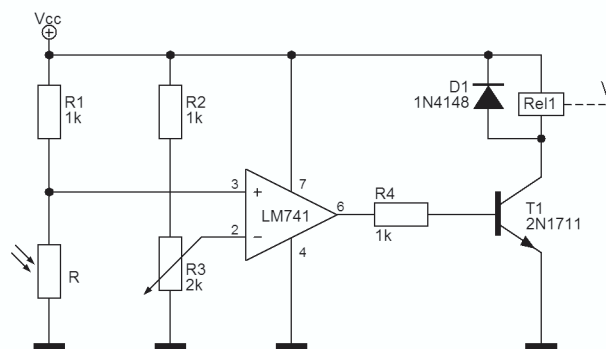
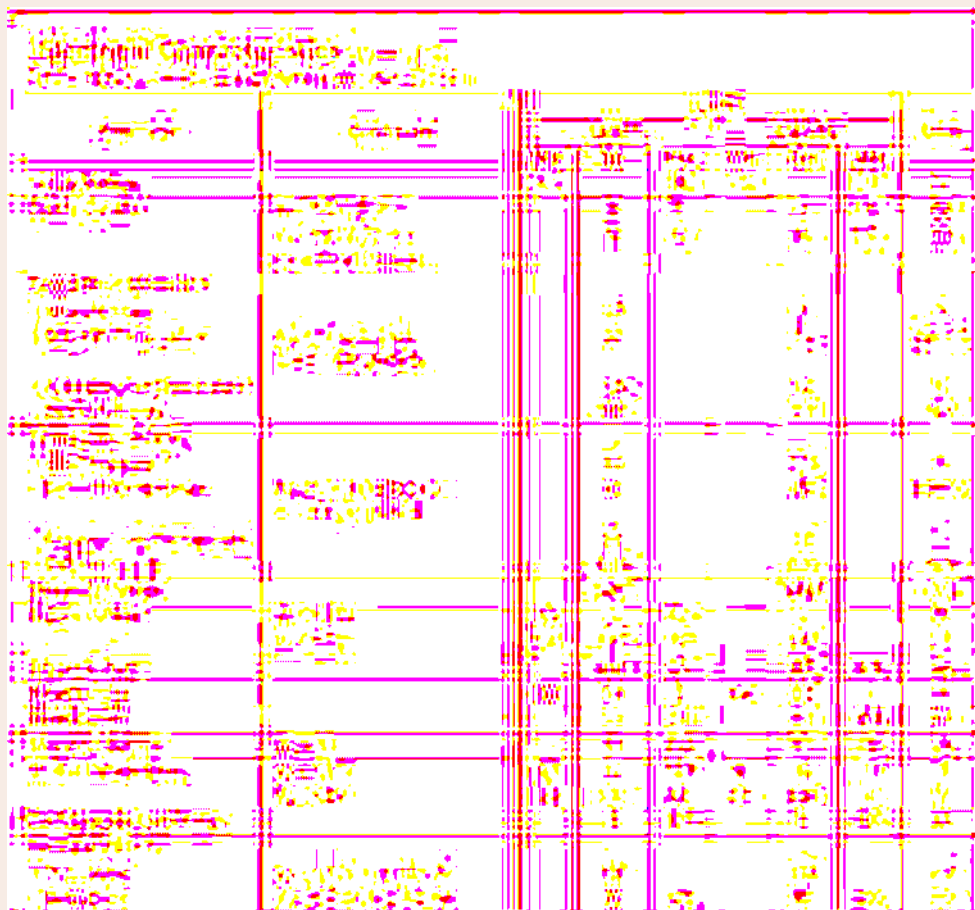
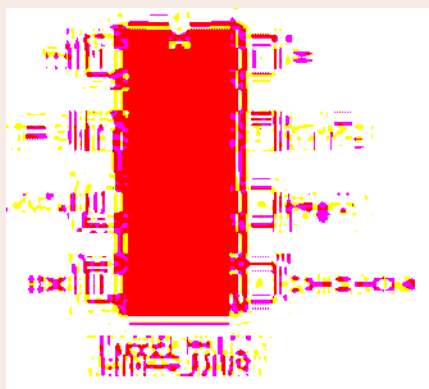



Figura 13: La soluzione proposta da Gianni di Firenze

transistor ed il diodo, come già spiegato qualche numero fa.
Il progetto di Gianni è perfetto, facciamo solo un appunto sulla scelta dei valore di R3. Il valore commerciale è 2.2KOhm e non 2KOhm pertanto la tensione di riferimento può variare da 0 a 0.68·Vcc (quindi 8,25V).
Ovviamente questo piccolo particolare non compromette il funzionamento del circuito.
Consiglio inoltre di inserire un condensatore da 10µF tra il morsetto invertente e massa in modo da limitare i disturbi sulla tensione di riferimento.
La soluzione di Gianni è stata scelta per la chiarezza con cui è stata descritta, ma devo fare i complimenti anche agli altri lettori che hanno inviato le loro proposte dalle quali traspare una vena di ingegno veramente innata!

LM555/LM555C TIMER

Le fiere e mostre mercato di Ottobre 2004

30 Settembre – 02 Ottobre 2004
11ª EDIZIONE SAT EXPO

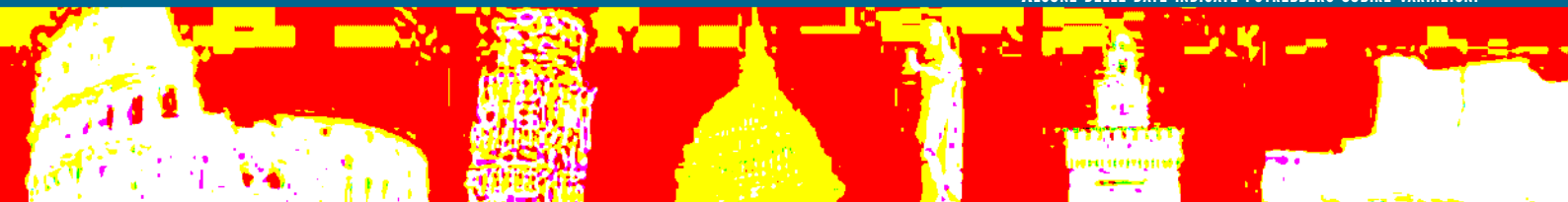
SAT Expo - Advanced Telecommunications è la vetrina di riferimento per le comunicazioni via satellite e per la Tv digitale terrestre, tecnologia che ha tenuto a battesimo con la prima trasmissione pubblica sperimentale in Italia e con la prima Conferenza nazionale dedicata al tema.
È una manifestazione che presta grande attenzione e spazio anche alle soluzioni wireless, all'automazione domestica, al cablaggio strutturato, quindi a tutte le più avanzate e innovative tecnologie di comunicazione oggi sul mercato o in fase di test.
Dal 2004, per la prima volta, è ufficialmente Salone internazionale: il riconoscimento, arrivato dall'organismo di controllo, proietta SAT Expo nei calendari e nei circuiti delle più prestigiose manifestazioni nel mondo.
Tra gli espositori* aziende leader come Eutelsat, British Telecom, Ses Astra, Sky Italia, Philips, Hughes, ESA (Agenzia spaziale europea), RAI, Telespazio, Telecom Italia, Globecast, Yamaha, Bticino, New Skies, Viasat.
I numeri della fiera: A SAT Expo 173 espositori da 20 Paesi diversi, su un'area fieristica di 30 mila metri quadri. 12.600 mila visitatori, di cui il 90% operatori professionali. *(dati riferiti all'edizione 2003)*

Luogo: Fiera di Vicenza
Orari: 9.00/18.00
Organizzatore: Promospace (www.satexpo.it)
Ingresso: Riservato agli organizzatori, previa registrazione

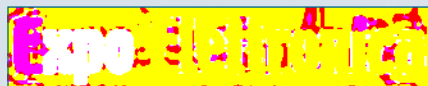
02-03 Ottobre 2004
RADIAN AND SILICON

La Mostra RADIANT è nata nel 1991. Inizialmente la Manifestazione era soprattutto rivolta al settore radiantistico con una componente dedicata alle radio d'epoca. Fin dalla prima edizione ha goduto del patrocinio dell'A.R.I., Associazione Radioamatori Italiani, diventando presto la più importante Mostra Mercato nazionale in questo campo.
Dal 2000 la Manifestazione ha intrapreso un deciso sviluppo con l'inserimento di nuove tematiche e l'assunzione di un sottotitolo particolarmente significativo: RADIANT AND SILICON - l'evoluzione della comunicazione e dal 2002 la periodicità è divenuta quadrimestrale con appuntamenti fissi nei mesi di gennaio, giugno e ottobre.
La Mostra comprende tra le altre le seguenti voci merceologiche: apparecchi e componenti per le telecomunicazioni, ricetrasmissioni, elettronica, informatica, videogiochi, surplus, editoria specializzata, radio d'epoca. Attualmente RADIANT, che si sviluppa su una superficie espositiva di 10.000 mq., conta più di 160 espositori ed oltre 12.000 visitatori.

Luogo: Segrate (MI)
Orari: Il 2 Ottobre 9.00/18.00
il 3 Ottobre 9.00/17.00
Organizzatore: Comis - Parco esposizioni Novegro (www.parcoesposizioninovegro.it)
Ingresso: n.d.

Alcune delle date indicate potrebbero subire variazioni

09-10 Ottobre 2004
FIERA DEL RADIOAMATORE

LUOGO: Zona Industriale Tito Scalo (PZ)
ORARI: n.d.
ORGANIZZATORE: EFAB srl (www.efab.it – Tel. 0971/485348)
INGRESSO: n.d.

16-17 Ottobre 2004
EXPO ELETTRONICA FAENZA

Da molti anni l'elettronica è entrata a far parte del nostro quotidiano, non solo in ambito professionale ma anche fra le mura di casa: dagli elettrodomestici ai giochi dei bambini, dagli antifurto alla Tv satellitare.
Expo Elettronica a Faenza propone una vasta panoramica sia sull'elettronica di consumo sia su quella rivolta ad un pubblico più esperto. Attrazioni: sala pose, mercatino dei privati. Settori commerciali: elettronica, computer, radiantismo, telefonia, radio d'epoca, fotografia, collezionismo. Espositori: 150.

LUOGO: Centro Fieristico Provinciale - Faenza (RA)
ORARI: 9.00/18.00
ORGANIZZATORE: Blu Nautilus srl (www.blunautilus.it)
INGRESSO: Intero € 7,00 - ridotto € 6,00

21-25 Ottobre 2004
SMAU 2004

Smau 2004 si svolgerà dal 21 al 25 ottobre a Fiera Milano e avrà come filo conduttore la convergenza: tra le reti, i servizi e le applicazioni. La manifestazione, che è la seconda al mondo nel settore hi-tech per numero di visitatori e metri quadrati di esposizione (nel 2003 380.000 visitatori e 80.000 metri quadrati), sarà strutturata in tre grandi settori di riferimento - business, consumer, e government e servizi al cittadino -, ospiterà anche Smau ILP, la più grande kermesse di videogiochi in Europa, Smau Shop e Smau Book Shop, dove sarà possibile acquistare i prodotti esposti nella parte fieristica e libri.

LUOGO: Fiera di Milano (MI)
ORARI: 10.00/19.00
ORGANIZZATORE: SMAU (www.smau.it)
INGRESSO: € 10,00 (gratuito con invito operatore)

23-24 Ottobre 2004
MERCATINO DEL RADIOAMATORE

LUOGO: Capannone Tecnobeton Zona Industriale Monopoli (BA)
ORARI: 9.00/13.00,16.00/20.00
ORGANIZZATORE: ARI Bari (080.5482374)
INGRESSO: n.d.

23-24 Ottobre 2004
MOSTRA DELL'ELETTRONICA

LUOGO: Centro Fieristico di Scandiano (RE)
ORARI: n.d.
ORGANIZZATORE: Comune di Scandiano (http://www.fierascandiano.it)
INGRESSO: n.d.

ALLA SCOPERTA DELL'OPTOELETTRONICA

quinta parte

LA FOTORESISTENZA

di Andrea Perilli
aperilli@infinito.it

*In questa puntata parleremo dell'unico componente passivo di tutta la rubrica: la fotoresistenza.
Tratterremo pertanto la composizione e il funzionamento di questo componente. Spiegheremo come si collega e dove impiegarlo. A fine articolo presenteremo un progetto pratico utile: un interruttore crepuscolare.*

LA TEORIA

COS'È UNA FOTORESISTENZA

Finora abbiamo trattato di tutti quei componenti che funzionando generano luce: i led, i led infrarossi, i led tricolori, i display e via dicendo.
In questa puntata invece, parleremo di un componente che riceve luce per funzionare: la fotoresistenza.
Già dal nome possiamo intuirne il funzionamento: "fotos" dal greco luce ci indica infatti che la resistenza del componente è relazionata alla luce.
A questo punto però, passateci il gioco di parole, è giunto il momento di far luce su questo componente.
La fotoresistenza è un dispositivo che varia il suo valore ohmico in relazione alla quantità di luce che lo investe. Per essere più precisi: è una resistenza variabile che presenta il suo massimo valore in totale oscurità che poi gradualmente si riduce con l'aumentare della luce

Figura 1: *Struttura di una fotoresistenza*

fino al minimo valore ohmico con una luminosità intensa. Naturalmente la diminuzione o l'aumento del valore resistivo non è lineare come vedremo più avanti, ma segue una curvatura caratteristica per ogni diverso componente.
Esistono in commercio diversi modelli di fotoresistenza che presentano svariati valori resistivi in piena oscurità.
La loro costituzione è pressoché identica anche se esternamente presentano involucri differenti.

Struttura di una fotoresistenza
Come mostra la figura 1 la fotoresistenza si compone di quattro parti:

- Serpentina conduttrice.
- Strato di solfuro di cadmio.
- Supporto isolante.
- Terminali.

La serpentina è formata da un materiale conduttore o in alcuni modelli da super conduttore a cui sono direttamente saldati i terminali, tutto poi protetto da una pellicola plastica.
Il cuore della fotoresistenza è lo strato di solfuro di cadmio (CdS) che in alcuni modelli è sostituito dal seleniuro di cadmio o dal solfuro di piombo.
Il supporto isolante è la base su cui è costruita la resistenza può essere di materiale plastico oppure di ceramica.
In pratica sul supporto isolante è poggiata la serpentina a cui è interposto uno strato di solfuro di cadmio che reagisce alla luce aumentando o diminuendo il passaggio di elettroni e quindi la resistenza ohmica.
Nella figura 2 è illustrato un grafico dove troviamo sull'ascissa la luminosità e sull'ordinata la resistenza ohmica. Osservando il grafico con attenzione notiamo un'unità di misura nuova: il Lux.

CHE COSA È IL LUX

Parlando di luminosità o per meglio dire di illuminamento bisogna parlare necessariamente del Lux.
Per definizione un Lux è l'illuminamento prodotto uniformemente dal flusso di un lumen su una superficie di un metro quadrato. Un lumen invece equivale circa ad una Candela (diciamo questo per non addentrarci troppo tra fisica e formule non gradite a tutti). A titolo informa-

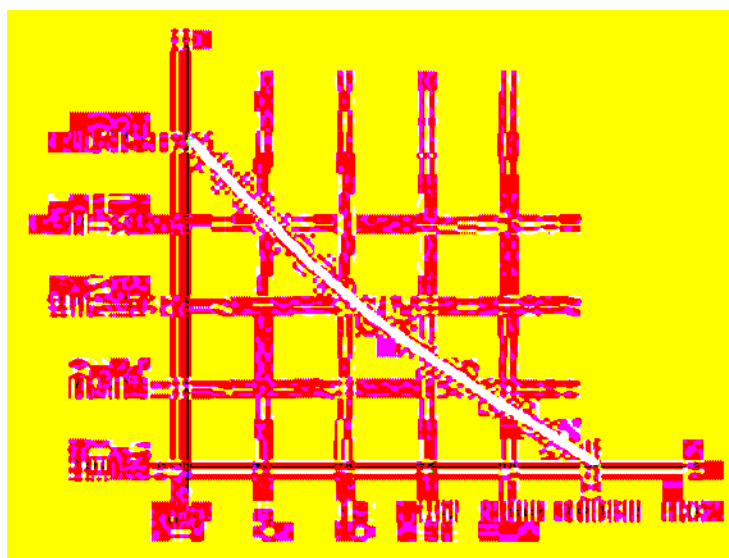



Figura 2: *Diagramma di risposta di una fotoresistenza in funzione della luminosità*

Simbolo	Unità nel SI	Misura	Definizione
E	lux (lx)	Illuminamento	Flusso luminoso incidente per unità di superficie
Φ	lumen (lm)	Flusso luminoso	Quantità di luce emessa nell'unità di tempo da una sorgente
I	candela (cd)	Intensità luminosa	Flusso luminoso emesso per angolo solido unitario
L	nit (nt)	Luminanza	Intensità di luce emessa per superficie unitaria emittente
Q	lumen secondo (lm * s)	Quantità di luce convenzionale	Flusso luminoso per tempo di illuminazione

Tabella 1: *Unità di misura fotometriche*

tivo riportiamo nella tabella 1 le unita di misura fotometriche.

COME SI COLLEGA UNA FOTORESISTENZA

La fotoresistenza essendo un componente passivo non necessita di particolari attenzioni alla polarità, infatti funziona in entrambi i versi. L'unica raccomandazione è di non collegare direttamente una fotoresistenza ad un carico poiché non sarebbe capace di resistere ad una corrente elevata. Il sistema corretto di impiegare una fotoresistenza è di accoppiarla ad un transistor oppure ad un amplificatore operazionale in modo da amplificare la sua debole corrente e poter pilotare un relé o un carico.
Nella figura 3 vi mostriamo uno schema applicativo di una fotoresistenza. Quando la luminosità che investe la fotoresistenza supererà il livello impostato dal Trimmer e più precisamente quando il valore ohmico della fotoresistenza abbatterà la forzatura a massa del Trimmer, allora il transistor condurrà accendendo la lampada (o il carico). In breve questo circuito si attiva con la luce.

Nella figura 4 invece otteniamo il funzionamento inverso: quando la luminosità sarà minore del livello impostato dal Trimmer o meglio quando il valore ohmico della fotoresistenza aumenterà cessando la forzatura a massa, il transistor finora interdetto, condurrà accendendo la lampada. In pratica questo circuito si attiva con l'oscurità.

LA PRATICA INTERRUTTORE CREPUSCOLARE

Chissà quante volte vi sarà capitato di dover uscire in giardino di sera e dopo essere scesi dover rifare le scale perché avevate dimenticato di accendere i lampioncini del vialetto?
Quante altre volte siete rincasati di tardo pomeriggio e avete trovato le luci del cortile già accese, perché lo erano dalla notte prima?
Meglio ancora, vostro figlio o fratellino vuole per forza dormire con una luce accesa ma quando è mattina dimentica di spegnerla?
Se purtroppo avete questi problemi o se semplicemente necessitate di un dispositivo che accenda delle luci di sera e le spenga di mattina, abbiamo ciò che fa per voi: un interruttore crepuscolare.

Figura 3: *Schema applicativo in cui la luminosità attiva il carico*

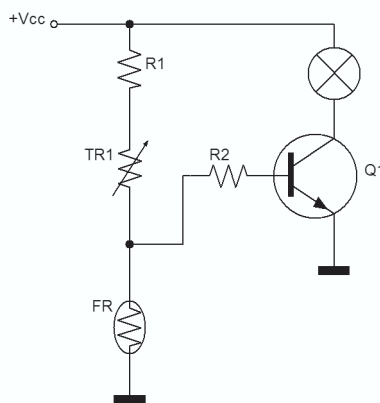



Figura 4: *Schema applicativo in cui l'oscurità attiva il carico*

COSA VUOL DIRE CREPUSCOLO?

Per definizione il crepuscolo è quello spazio di tempo antecedente al sorgere del sole o appena successivo al tramonto, in cui i raggi solari, che penetrano nell'atmosfera con una inclinazione maggiore, producono quella tipica luminosità detta appunto crepuscolare.
Esistono due fasi crepuscolari: la fase mattutina e la fase serale detta anche vespertina. Le due fasi si distinguono tuttavia in tre categorie: crepuscolo civile, quando il sole si trova 6° sotto l'orizzonte; crepuscolo nautico, sole a 12° e infine crepuscolo astronomico con il sole a 18°. Naturalmente la durata dei fenomeni varia a seconda della latitudine dal minimo, all'altezza dell'equatore, fino alla sua massima durata salendo verso i poli terrestri.
Detto ciò si deduce facilmente che un interruttore crepuscolare è un dispositivo che opera durante questo fenomeno. Precisamente il dispositivo si

Figura 5: *Foto della scheda a montaggio ultimato*

Figura 6: *Foto di un crepuscolare che pilota l'accensione della illuminazione pubblica*

attiva nel crepuscolo serale e si disattiva al crepuscolo mattutino.
Il principio di funzionamento è molto semplice: la fotoresistenza, come abbiamo appreso nella teoria, varia il suo valore ohmico al variare della luminosità che la investe. Se quindi riusciamo a fare in modo che ad un detto grado di luminosità la fotoresistenza attivi un carico, abbiamo realizzato un interruttore crepuscolare.

I pregi del crepuscolare

Vi starete chiedendo perché realizzare un interruttore crepuscolare quando invece basterebbe dotare le luci di un programmatore orario?
Possiamo rispondere con due semplici ragioni:

1. Il crepuscolare una volta regolato funzionerà sempre alla stessa luminosità senza risentire dell'ora legale o solare.
2. Durante i temporali diurni magari anche con nebbia il crepuscolare potrebbe attivarsi comunque mentre un programmatore orario dovrebbe per forza essere azionato manualmente.

Se ciò non bastasse possiamo dirvi che gran parte delle illuminazioni pubbliche sono attivate tramite interruttori crepuscolari. Infatti se avete la fortuna di osservare vicino alle cabine elettriche della pubblica illuminazione vi troverete una scatola plastica di colore bianco che pilota direttamente la bobina del contattore nella cabina.

SCHEMA ELETTRICO

Lo schema elettrico di figura 7 è molto intuitivo e come i precedenti è didatticamente molto basilare.
Il cuore del circuito è U1 l'ormai noto amplificatore operazionale quad LM 324 configurato come comparatore di tensione.
All'ingresso non invertente troviamo il partitore formato da R1 e TR1 che servono a impostare la tensione di riferimento del comparatore ma in pratica TR1 regola la soglia di luminosità alla quale il dispositivo si attiva.
All'ingresso invertente troviamo la fotoresistenza e R2, al variare della luminosità varierà anche il valore di FR1. Quando il valore di tensione all'ingresso non invertente sarà maggiore della tensione sull'ingresso invertente, l'uscita del comparatore si porterà a livello alto: il valore della Vcc. All'uscita (pin 1) è collegata la resistenza R3 che polarizza la base del transistor Q1 che a sua volta attiva il relé e di conseguenza il carico ad esso collegato.
Il diodo D1 serve a proteggere il transistor da disturbi prodotti dall'eccitazione della bobina del relé. Il condensatore C1 serve a evitare che lampi di luce improvvisi facciano disattivare il carico. Abbiamo inserito questo piccolo accorgimento per evitare che per esempio durante qualche temporale i fulmini avessero potuto far spegnere le luci del vialetto oppure per evitare che se le luci dell'auto in manovra avessero colpito accidentalmente la fotoresistenza avrebbero causato lo spegnimento del carico.C1 induce nel circuito un ritardo nella risposta di circa tre secondi sufficienti a non far disattivare il carico con improvvisi lampi di luce (naturalmente questo tempo può variare a seconda della tolleranza del condensatore e della resistenza della bobina del relé).

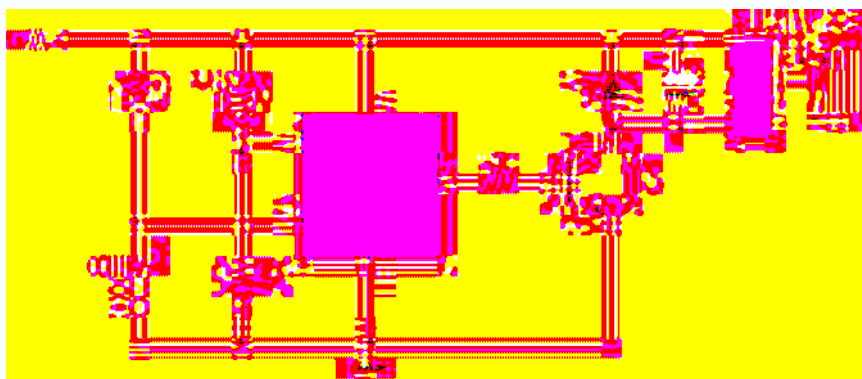



Figura 7: Schema elettrico

Elenco componenti	
Sigla	**Valore**
R1, R3	1 KΩ 1/4 W
R2	10 KΩ 1/4 W
R4	100 KΩ 1/4 W
TR1	Trimmer multigiri orizzontale da 100 KΩ (RS 186-081)
D1	1N4007
C1	1000 μF elettrolitico
U1	LM324
FR1	Fotoresistenza
Q1	BC547
RL	Relé NAIS 12 V doppio scambio
Zoccolo per integrato a 14 pin	
Morsetto serrafilo 2 vie	
Morsetto serrafilo 3 vie	

REALIZZAZIONE PRATICA

Dopo aver realizzato il circuito stampato servendovi del master di figura 8 possiamo procedere al montaggio dei componenti partendo da quelli a basso profilo. Cominciamo dalle tre resistenze, il diodo D1 per seguire poi con lo zoccolo dell'integrato orientandolo come mostra il piano di montaggio di figura 9. Successivamente salderemo il Trimmer, il transistor, il condensatore C1, e in ultimo il relé. Infine potrete dotare il circuito del morsetto serrafilo per il collegamento dei fili del carico.

COLLAUDO E TARATURA

Per effettuare il collaudo avremo bisogno di un alimentatore a 12 v e di un carico da collegare al relè, sia esso una lampadina o un semplice diodo led.
Dopo aver dato tensione al circuito bisognerà regolare il Trimmer al suo massimo valore. A questo punto abbiamo due alternative: attendere il crepuscolo o operare in maniera artificiale. Il sistema più attendibile sarebbe comunque attendere il crepuscolo così da regolare il Trimmer fino all'accensione del carico con la luminosità desiderata, sicuri che ogni sera il circuito agisca con quella stessa luminosità. Ammesso però di non voler attendere, dopo aver coperto la fotoresistenza al grado di oscurità voluta occorrerà ruotare il perno di TR1 fino a quando il carico si attiverà.

Figura 8: Circuito stampato scala 1:1 (lato rame)

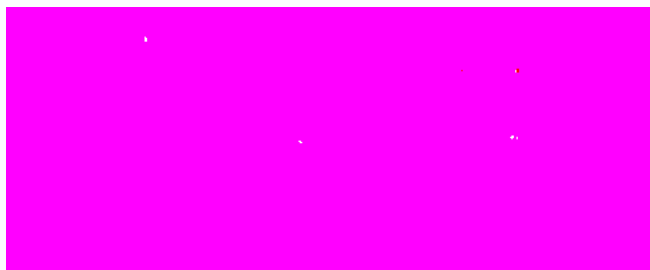



Figura 9: Piano di montaggio della scheda

Alcuni consigli utili

1 Come al solito se collegherete al relé un carico a 240 volt dovrete inscatolare il circuito per non rischiare pericolose

scosse.

2 Quando installate il circuito fate il modo che la fotoresistenza capti solo la luce ambientale.
3 Collocate la fotoresistenza in modo che non venga investita dalla luce del carico oppure otterrete un ottimo lampeggiatore.

A PROPOSITO DI LAMPEGGIATORI

Per farvi un piccolo esempio: notate quei lampeggiatori a batteria che si usano per evidenziare zone pericolose nei lavori in corso (figura 10), non possiamo dirvi come sono fatti per motivi di brevetto ma possiamo spiegarvi il principio di funzionamento.

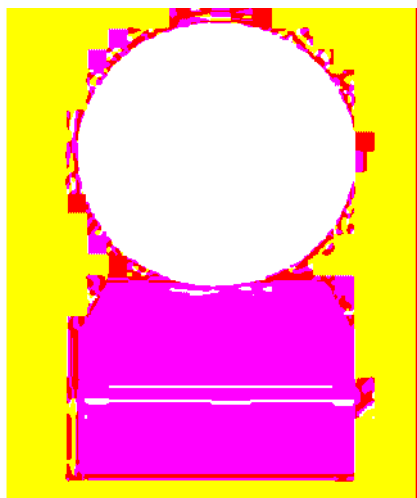
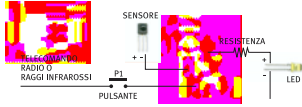
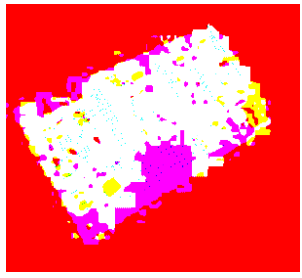
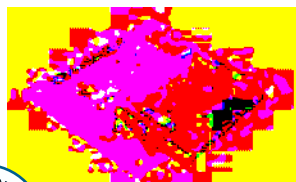



Figura 10: *Disegno di un lampeggiante per aree pericolose*

Il circuito interno è un interruttore crepuscolare, infatti di giorno i lampeggiatori sono inattivi.
Appena fa sera il crepuscolare entra in funzione, il sensore che finora aveva captato l'oscurità ambientale riceve ora la luce del lampeggiante, "credendo" quindi che sia giorno disattiva il lampeggiante ma captando di nuovo l'oscurità riaccende la lampada.

Il ciclo si ripete fino a che la luce ambientale e quindi il giorno, disattivano il crepuscolare. Capite quindi perché la fotoresistenza non dovrà mai essere investita dalla luce del carico.

SISTEMA DI MISURA 1-WIRE

seconda parte

INSTALLAZIONE DEL SOFTWARE

di Nicola Ulivieri e Luca Tondi
nicola@ing.unisi.it

In questo articolo continueremo lo sviluppo del progetto del sistema 1-Wire ed in particolare, come vi avevamo preannunciato nel numero precedente, descriveremo dettagliatamente le fasi necessarie per l'installazione del software che consente la gestione dell'interfaccia USB-1-Wire e di tutti i dispositivi (termometri, convertitori, memorie, potenziometri digitali, switches) che vi si possono connettere. I programmi di interfaccia necessari per garantire il funzionamento del sistema sono gratuiti e non necessitano di alcuna licenza.

Coloro che hanno realizzato il dispositivo di interfaccia 1-Wire per PC, proposto nel numero precedente, non vedranno certo l'ora di poterlo utilizzare almeno con il primo dispositivo che vi abbiamo consigliato, il termometro 1-Wire DS18S20.
Seguendo le istruzioni che vi daremo sarete presto in grado di veder comparire sul vostro schermo il grafico con l'andamento della temperatura misurata ogni secondo dal DS18S20.
Premettiamo innanzi tutto che l'installazione si riferisce ad un computer con sistema operativo Windows 2000 o XP.

INIZIAMO L'INSTALLAZIONE

Se avete visitato il sito web della Dallas/Maxim (www.maxim-ic.com), avrete notato la discreta quantità di software che viene messa a disposizione degli utenti di dispositivi 1-Wire. Ci sono programmi in C, Visual Basic ed altro, che possono essere utilizzati e modificati per creare degli eseguibili per le nostre specifiche esigenze. Naturalmente questo comporta una certa conoscenza sia dei linguaggi sia dei dispositivi che stiamo utilizzando. Per chi non ha particolari esigenze, ma vuole semplicemente interagire con i dispositivi, è stata sviluppata in Java l'applicazione *OneWireViewer*, ed è quella ci apprestiamo a descrivervi. Per prima cosa dovete scaricare dal sito di Fare Elettronica il file relativo a questo articolo e poi…

Java Innanzi tutto

Se avete il *Java Runtime Environment* versione *1.4.0* o successiva, già installato sul vostro PC, siete a posto, altrimenti dovete scaricare il file *j2re-1_4_2_04-windows-i586-p.exe* dal sito *Java.Sun*, operazione un po' "faticosa" se vi connettete da casa e non avete a disposizione una linea veloce (14.52MB!).
Tale file istallerà la versione *1.4.2* (l'ultima al momento disponibile). Per il download procedete così: andate sul sito *http://java.sun.com/j2se/1.4.2/download.html*, e cliccate al cen-

tro su *Download J2SE JRE*; si aprirà una pagina in cui dovrete accettare i termini della licenza, dopodiché apparirà la pagina con il link al file.
Fate attenzione a non scaricare l'altro file più piccolo da 1.35MB in quanto questo esegue l'installazione on-line, cioè mentre siete connessi a Internet. Una volta salvato il file in una cartella a piacere (per esempio.: *c:\temp*) avviatelo. Vi verrà chiesto di accettare i termini della licenza e poi il tipo di installazione; scegliete quella tipica (figura 1). Al termine, se non avete *Windows XP*, vi verrà chiesto di riavviare il computer. A questo punto possiamo proseguire con l'installazione dei drivers per il 1-Wire.

Figura 1: *Scelta del tipo di installazione del Java Runtime Environment*

Per il Plug & Play
Ora aprite il file *Windows.zip* al cui interno sono contenuti tre files che dobbiamo copiare nelle cartelle (o *directory*) indicate nella tabella1. Può darsi che in alcune versioni di Windows non sia presente la cartella *C:\windows* ma che al suo posto possiate trovare la cartella *C:\WinNT* che comunque è perfettamente equivalente. Se non trovate la sottocartella *\inf*, cioè la cartella *C:\windows\inf*, potete crearla voi senza pericolo di commettere alcun errore. In realtà, tutto questo dovrebbe essere automaticamente eseguito dal programma di installazione che vi descriveremo tra breve; tuttavia, visto che nelle varie prove eseguite, talvolta sono stati riscontrati dei problemi nel riconoscimento automatico, abbiamo preferito segnalarvi come eseguire la procedura manualmente. Ed ora eccoci finalmente pronti per il primo test.

Connessione e riconoscimento
A questo punto potete connettere il vostro modulo di interfaccia al PC (figura 2) che dovrà rilevare la presenza del nuovo hardware e vi richiederà l'installazione del software per l'*USB Host 1-Wire* (figura 3).

Cliccate sulla prima scelta relativa all'installazione automatica ed andate avanti. Windows cercherà il software di installazione ma con esito negativo; infatti è necessario specificare il percorso per raggiungere il file

Nome	Copiare in
DS2490.sys	C:\WINDOWS\system32
DS24902k.inf	C:\WINDOWS\inf
DS24902k.PNF	C:\WINDOWS\inf

Tabella 1: *Estrazione dei files contenuti in Windows.zip*

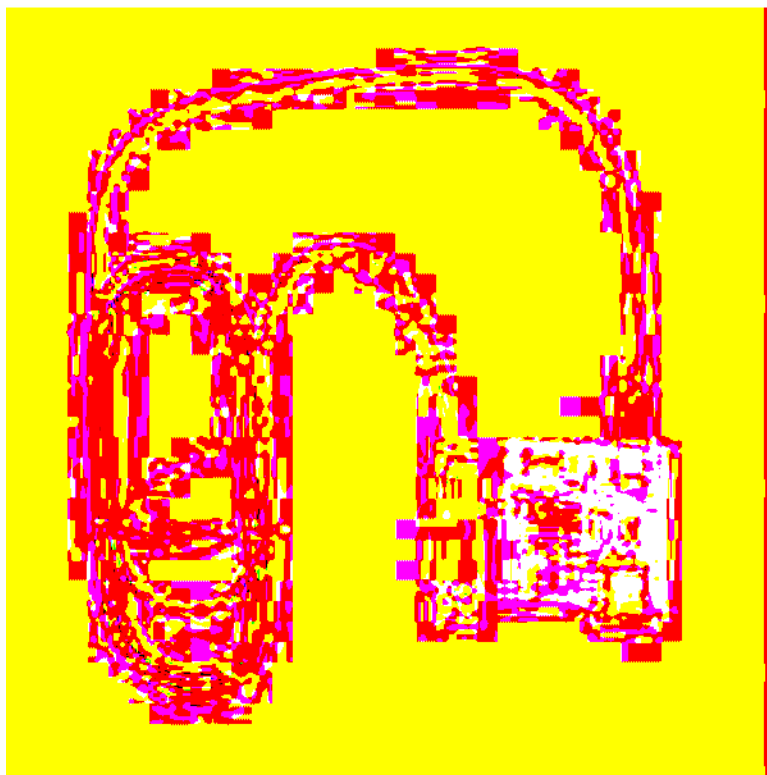



Figura 2: *Il modulo di interfaccia, presentato il mese precedente, connesso al cavo USB*

Figura 3: Riconoscimento nuovo hardware e installazione del software

DS2490.sys che avevamo salvato in *C:\WINDOWS\system32* (figura 4). Una volta risposto a questa domanda vi apparirà la finestra con la conferma di corretta installazione del software.
In alcune versioni di Windows può capitare che venga richiesto di specificare anche la posizione del file DS24902k.inf; In tal caso ricordatevi che è *C:\windows\inf*.

Figura 4: Richiesta del file DS2490.sys

Diversamente dalle varie periferiche di archiviazione gestite dalla porta USB, come le memorie o le "penne USB", il modulo 1-Wire non appare tra le icone attive della barra delle applicazioni. Verifichiamo quindi la presenza del nostro modulo in un altro modo: sul desktop, cliccate su *"Risorse del computer"* con il tasto destro del mouse e scegliete *Proprietà*. Sulla finestra che appare, selezionate *Hardware* (in alto al centro della finestra) e quindi *Gestione Periferiche* (o *Device Manager* se avete la versione Inglese - figura 5). La prima tipologia di dispositivi della lista che compare sarà proprio la *1-Wire devices*, sotto cui troverete la vostra *USB Host for 1Wire Microlan*. Per verificare il corretto funzionamento del dispositivo potete cliccare con il destro del mouse su questa voce e scegliere *Proprietà* (figura 5). La finestra che appare vi confermerà che il dispositivo sta funzionando correttamente e quindi che abbiamo fatto il primo passo verso il riconoscimento automatico del nostro sistema di misura.

Altri drivers

All'interno del file compresso *Drivers v400b3.zip* è contenuto il programma *Install_1_Wire_Drivers_v400b3.exe*, che deve essere copiato in una cartella qualsiasi (ad esempio in *C:\temp*) ed avviato; a questo punto vi verrà chiesta la conferma e verrà avviata l'installazione di cui dovete seguire attentamente le indicazioni (figura 6).
Come potete osservare dalla figura 7, è necessario scollegare il modulo, accettare i termini previsti dalla licenza del software e infine selezionare la cartella dove si vogliono installare i driver; vi consigliamo di lasciare quella di default.
Terminata l'installazione vi verrà chiesto di connettere nuovamente il modulo alla USB per terminare la configurazione (figura 8).

Alla comparsa della finestra mostrata in figura 9, premete il pulsante *Auto-Detect* in basso a sinistra e ignorate il messaggio

Figura 5: Verifica del riconoscimento del modulo Host 1-Wire

Figura 6: *Installazione dei drivers completata*

Figura 7: *Richiesta di scollegamento del modulo di interfaccia*

Figura 8: *Richiesta di collegamento del modulo di interfaccia*

di avvertimento successivo. Verrà avviata la scansione di tutte le porte alla ricerca della presenza del modulo adattatore e al suo rilevamento vi verrà chiesto se volete configurarlo come predefinito (figura 10).

Figura 11: *Files istallati dal programma Install_1_Wire_Drivers_v400b3.exe.*

Poiché non abbiamo ancora altri moduli adattatori, scegliete naturalmente *Sì*.
Questa parte dell'installazione è conclusa e sullo schermo comparirà la finestra di figura 11, contenente alcuni files:

- il *ReadMe.htm* con le informazioni sui driver istallati;
- il *Default 1-Wire Net* per la scelta dell'adattatore (figura 9), nel caso in cui lo si voglia cambiare;
- il *.NET 1-Wire API Setup* per l'uso delle API (Application Program Interface); richiede l'istallazione del .NET Framework 1.1.4322 ma non verrà utilizzato per le nostre applicazioni;
- ed infine il *OneWireViewer.htm*, che è il file che andremo ad aprire per terminare l'installazione.

Ci siamo: installiamo l'osservatore 1-Wire

Avviando il file *OnWireViewer.htm* si aprirà una pagina html tramite il vostro browser predefinito (figura 12), con la quale possiamo avviare l'installazione (finalmente è l'ultima!) del programma in Java per la gestione dei dispositivi 1-Wire. Per eseguire questa operazione e terminare quindi l'installazione è necessario connettersi ad Internet. Cliccate in alto al centro sulla scritta CLICK HERE. Verrà ricercata in rete la versione più aggiornata del OneWire Viewer (figura 12), scaricata (figura 13) ed avviata dopo la vostra conferma.

Figura 9: *Finestra di dialogo per la selezione dell'adattatore 1-Wire*

Figura 10: *Scelta dell'adattatore predefinito*

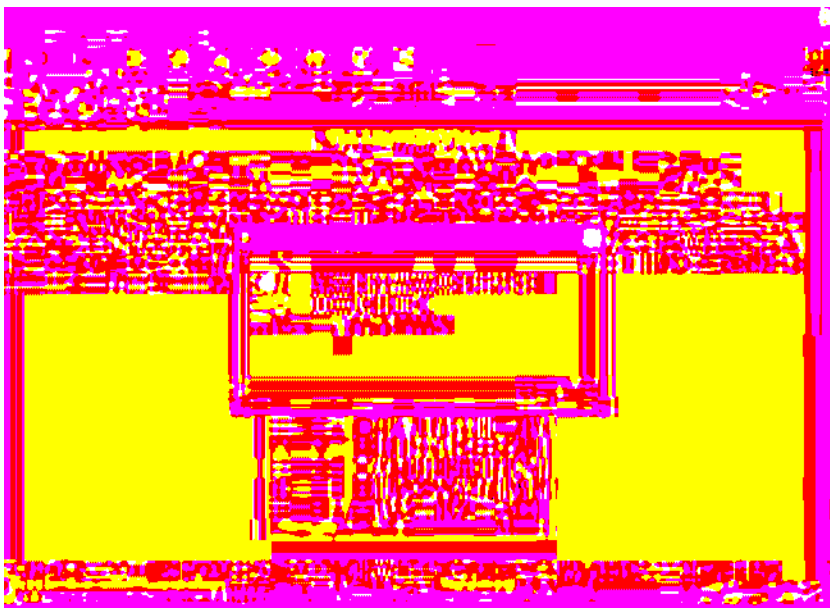



Figura 12: *Alla pressione del collegamento CLICK HERE, verrà cercata in rete la versione più recente del OneWire Viewer*

Figura 13: *Download della versione più recente del OneWire Viewer.*

Figura 14: *Richiesta di creazione dell'icona per lo shortcut all'applicazione OneWireViewer*

Alla richiesta se si desidera integrare "OneWireViewer Application" nell'ambiente Desktop rispondete di sì (figura 14); in questo modo verrà creata un'icona sul desktop che vi permetterà di avviare agevolmente il programma di controllo.

Al primo avvio, sarà necessario specificare il tipo di adattatore e la porta USB utilizzata (figura 15), la velocità con la quale

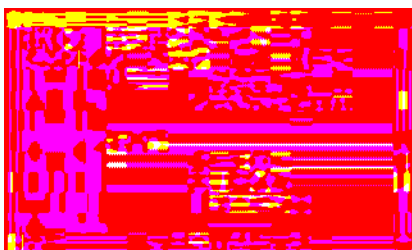



Figura 15: *Scelta della porta dell'adattatore*

Figura 16: *Scelta della velocità di polling*

Figura 17: *Scelta della visualizzazione dei dispositivi etichettati*

interrogare i dispositivi connessi (*Polling Rate* - figura16) e se vogliamo visualizzare dispositivi con etichetta (*tagged* figura 17). Queste impostazioni possono essere comunque cambiate in seguito, semplicemente.

L'installazione del software è terminata e la comparsa della finestra del *OneWireViewer* mostrata in figura 18 ve ne darà conferma.

IL ONEWIREVIEWER

Sulla sinistra del pannello del *OneWireViewer* (figura 18) viene mostrata la lista dei dispositivi connessi all'interfaccia dei quali viene segnalato l'indirizzo e il nome. Ovviamente, se nessun dispositivo è connesso la finestra rimarrà vuota. Se non lo avete ancora fatto, collegate i due termometri DS18S20 (le modalità di collegamento sono state ampliamente descritte nella puntata precedente) e verificate

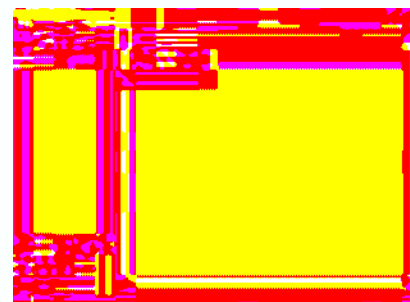



Figura 18: *Pannello del OneWireViewer*

come questi vengano automaticamente riconosciuti dal sistema che ne indica la presenza aggiornando la "Device List".

Se si seleziona uno dei dispositivi connessi, nella parte destra della finestra "OneWireViewer" (figura 19) si attiva automaticamente la finestra "Description" dove è possibile trovare tutte le informazioni relative a quel particolare dispositivo, come l'indirizzo, il nome, il nome alternativo e una breve descrizione delle caratteristiche operative.

Nel caso del termometro DS18S20, oltre alla finestra "Description" è presente anche la finestra "Temperature" che può essere selezionata con il mouse e che consente la visualizzazione in un grafico della temperatura rilevata dal sensore (figura 20). È possibile scegliere tra una misura in gradi

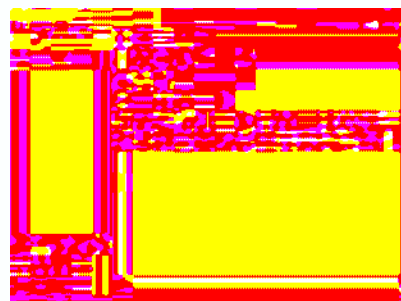



Figura 19: *Descrizione del dispositivo selezionato*

Figura 20: *Visualizzazione del grafico della temperatura*

Figura 21: *Finestra separata dedicata alla visualizzazione della temperatura acquisita da un dispositivo DS18S20; posizionando il cursore sul grafico e premendo il tasto destro, è possibile copiare i dati acquisiti negli appunti.*

Centigradi o Fahrenheit e tra una risoluzione di 0.1°C o 0.5°C. È possibile anche visualizzare il grafico in una finestra separata; per far questo basta andare sul menu "View" e scegliere "Show Tab in New Window" (figura 20) e verrà aperta una nuova finestra come mostrato in figura 21. Questo è particolarmente utile quando si hanno molti dispositivi e si vogliono visualizzare tutte le risposte su video. Una funzione particolarmente utile, offerta dal *OneWireViewer* è che i valori di temperatura misurati possono essere copiati semplicemente posizionandosi sul grafico, premendo il tasto destro del mouse e scegliendo il formato di separazione dei dati (figura 21). Dopo questa semplice operazione i dati possono essere riportati in un qualsiasi documento, come per esempio su Excel, per elaborarli o per personalizzare il grafico secondo le proprie esigenze (figura 22).

Figura 22: *Esempio della copia e visualizzazione dei dati di temperatura acquisiti con il OneWireViewer.*

Se volete modificare le impostazioni scelte inizialmente, come l'intervallo di campionamento, basta andare su "Tools", "Device Poll Rate" (figura 23) e cambiarle. La scelta dell'*Overdrive Mode* ("Tools", "1-Wire Speed", figura 23) consente un velocità di trasmissione dei dati circa 7 volte maggiore e può essere scelta solo se i dispositivi non sono molti e la lunghezza del bus non è superiore a qualche metro. Andando sul menu "View" e poi cliccando su "Show Message Log" si apre una finestra in cui possiamo vedere se ci sono errori di comunicazione con i dispositivi.

Figura 23: *Scelta della velocità di polling, velocità di trasmissione e tipo di adattatore*

Anche questa parte è terminata. Sperando che sia stata di vostro interesse, vi diamo appuntamento al prossimo mese per la puntata in cui spiegheremo come aggiungere altri due dispositivi al nostro sistema, un convertitore AD e una memoria EEPROM.

GLI STABILIZZATORI DI TENSIONE

quarta parte

di Nico Grilloni
nicogrilloni@tin.it

Dopo i classici e sempre validi regolatori a BJT, si passa adesso alla descrizione e alle modalità di dimensionamento dei regolatori a circuito integrato

Nei precedenti articoli sono stati illustrati i più classici stabilizzatori di tensione di tipo serie nei quali il componente di regolazione, al pari dello stadio amplificatore di errore, è realizzato tramite BJT. Crediamo opportuno ribadire che questi regolatori funzionano egregiamente e hanno buona affidabilità. Non è infatti casuale che prima dell'avvento dei circuiti regolatori integrati fossero comunemente utilizzati in tutte quelle configurazioni circuitali che, per un corretto funzionamento, avevano bisogno di una tensione costante sia nei confronti delle sempre possibili fluttuazioni della tensione di linea – che è quasi sempre la sorgente primaria di tensione – sia nei confronti delle altrettanto possibili, e spesso frequenti, fluttuazioni della corrente assorbita dal carico.

Il successivo ingresso nel mercato della componentistica dei regolatori integrati ha subito avuto un enorme successo. Infatti, sebbene il principio di funzionamento si basi quasi interamente sugli analoghi circuiti a componenti discreti (e per convincersi di ciò è sufficiente analizzare il circuito analogico corrispondente a un qualsiasi regolatore integrato), dal punto di vista pratico presentano notevoli vantaggi dei quali si citano i principali:

1. Alcuni tipi sono già predisposti per una notevole gamma di tensioni fisse (5 V, 6 V, 8 V, 10 V, 12 V, 15 V, 18 V e 24 V) con correnti da 100 mA a 5 A (e anche più elevate).
2. Nella maggior parte dei casi i regolatori integrati per tensioni fisse possono essere resi a tensione di uscita variabile con pochi componenti passivi esterni.
3. Esiste una vasta gamma di regolatori integrati già progettati per tensioni di uscita variabili e, queste ultime, sia positive che negative.
4. Consentono la realizzazione di stadi stabilizzatori di tipo tracking (ossia con tensioni positive e negative del medesimo valore rispetto a massa) di frequente utilizzazione, per esempio, nelle configurazioni che, fra i vari componenti attivi, comprendono gli amplificatori operazionali o altri componenti attivi che necessitano di alimentazione positiva e negativa del medesimo valore.
5. Sono dotati di circuito shuntdown che ne blocca il funzionamento nei casi di sovratemperatura o di sovracorrente.
6. Sono di facile dimensionamento.
7. Presentano minimo ingombro. Le serie più recenti hanno dimensioni di qualche millimetro quadrato per qualche millimetro di altezza.

Da queste caratteristiche si evincono facilmente i motivi del successo dei regolatori integrati.

Figura 1: *Regolatore a tre terminali. Si noti l'estrema semplicità circuitale. Per la realizzazione sono infatti sufficienti soltanto due condensatori*

I REGOLATORI A TRE TERMINALI CON V_o FISSA

Hanno solo il pin di ingresso a cui si applica la tensione raddrizzata e filtrata da un condensatore di opportuna capacità (v. articoli precedenti), il pin di uscita e il pin che va connesso a massa. La serie più nota ha sigla 78XX ed è disponibile sia in involucro plastico che metallico del tutto simile, quest'ultimo, all'involucro TO-3 dei BJT di potenza, mentre la serie 78LXX si presenta, se in involucro metallico, come un BJT di piccola potenza.

Un semplice esempio applicativo è esposto nella figura 1. Si può constatare come oltre all'IC siano presenti solo due condensatori. Il primo di questi, C1, è opportuno inserirlo solo nel caso in cui lo stadio regolatore non sia cablato nelle immediate vicinanze dello stadio raddrizzatore già dotato di filtro capacitivo (per immediate vicinanze i costruttori specificano una distanza non superiore a 6 pollici corrispondenti a circa 15 cm). Il secondo condensatore, C2, ha solo la funzione di tenere bassa l'impedenza di uscita dello stadio e di migliorare la risposta alle variazioni istantanee della corrente richiesta dal carico.

I parametri più importanti di questi regolatori sono la caduta di tensione ingresso-uscita (dropout voltage) e la corrente IQ assorbita indipendentemente dalla corrente richiamata dal carico (corrente di riposo o quiescent current).

Al fine di ottenere una buona regolazione, la tensione di dropout della serie 78XX dev'essere non inferiore a 2 V, mentre la corrente di riposo è di circa 5 mA (3 mA per la serie 78LXX). Si fa presente fin d'ora che in commercio esistono i regolatori LDO (Low Dropout) che presentano cadute di tensione ingresso-uscita estremamente basse, anche dell'ordine del decimo di volt, e correnti di riposo dell'ordine del microampere e anche minori. Il regolatore LP2950, per esempio, ha una tensione di dropout di circa 0,4 V e una corrente di riposo inferiore a 70 μA.

I REGOLATORI A TRE TERMINALI CON V_o VARIABILE

Pur potendosi realizzare regolatori a tensione di uscita variabile con la serie 78XX, per un'ottimale stabilità di funzionamento si preferisce ricorrere ai regolatori integrati a tre terminali già previsti dal costruttore per tensione V_o variabile.

Il più versatile regolatore attualmente nel mercato della componentistica è l'LM117 della National che, nelle sue diverse versioni (e sigle), consente tensioni di uscita variabili con continuità da 1,2 V a 40 V con una corrente di uscita di 1, 5 A. Le ulteriori, significative caratteristiche sono:

- Massima tensione in-out di 40 V.
- Tensione di riferimento, coincidente con la minima V_o, pari a 1,25 V.
- Protezione a limitazione di corrente. L'intervento, per il modello da 1,5 A, è a 2,2 A.
- Protezione dalle sovratemperature (shut-down termico).
- Punto di lavoro costante perché sempre all'interno della safe area protection.
- Potenza dissipabile in involucro TO-3 pari a 20 W.
- Tensione di rumore in uscita (rms in % della V_o), 0,003 %.
- Regolazione sul carico dello 0,1 % per una corrente di carico di 1,5 A.
- Regolazione di linea dello 0,05 % / V.

I terminali dell'LM117 sono il pin

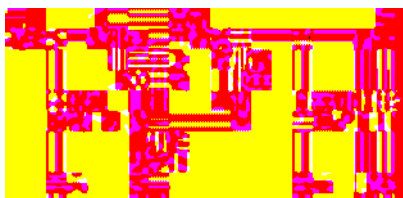



Figura 2: *Con il regolatore integrato della serie LM117 si possono realizzare sia stabilizzatori a tensione di uscita fissa che a tensioni di uscita variabile. Rispetto al circuito precedente sono state aggiunte soltanto le resistenze R_1 e R_2*

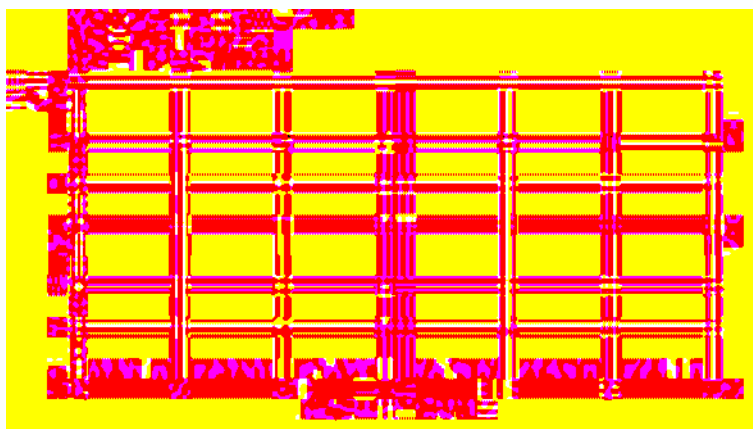



Figura 3: *Il marker orizzontale c indica la tensione di uscita, V_o = 20 V, del regolatore di cui alla figura 2*

in di ingresso, il *pin out* di uscita e il pin ADJ (adjustment). La figura 2 mostra la configurazione circuitale del regolatore che utilizza, appunto, l'LM117. Si constata come al regolatore di cui alla figura 1 si siano soltanto aggiunte due resistenze, la R_1 e la R_2.
L'espressione della tensione di uscita (che qui non dimostriamo restando comunque a disposizione del Lettore che volesse sapere come alla medesima si perviene) è:

$$V_o = V_{REF} \times \left(1 + \frac{R_2}{R_1}\right) = R_2 \cdot I_Q$$

In questa, *VREF* è la tensione di riferimento che per l'LM117 è pari a circa 1,25 *V*, e I_Q è la corrente di riposo che vale 50 μA. Per il basso valore di quest'ultima l'espressione precedente può porsi nella forma:

$$V_o = V_{REF} \times \left(1 + \frac{R_2}{R_1}\right)$$

Da quest'ultima espressione si vede come sia possibile, agendo sul valore delle due resistenze R_1 e R_2, rendere variabile la tensione V_o di uscita. Poiché il valore della resistenza R_1 è in genere fissato a 120 Ω o a 240 Ω, si agirà sul valore da attribuire alla R_2 per avere la V_{omax} desiderata. Infatti, dalla stessa espressione si vede subito che per R_2 = 0 la tensione di uscita coincide con il valore della tensione V_{REF} di riferimento, mentre il valore massimo della stessa V_o dipenderà solo dal valore della R_2 che, in pratica, si realizzerà tramite un trimmer di opportuno valore, oppure tramite un trimmer atto a fissare la massima V_o con in parallelo un potenziometro lineare, sul quale si agirà per variare la tensione di uscita dal valore minimo V_o = V_{REF} al valore massimo desiderato. In merito alla V_{omax} è solo da tener presente che la tensione V_i di ingresso, raddrizzata e filtrata, dev'essere di almeno 2 V maggiore della stessa V_{omax}. Ciò al fine di ottenere una regolazione ottimale.

Esempio
Si vuole dimensionare uno stadio del tipo illustrato nella figura 2 per una tensione di uscita V_o variabile da 1,5 V a 20 V.
Il dimensionamento dello stadio, impostata la V_i a non meno di 22 V, si esaurisce nel calcolo del valore massimo che deve avere il trimmer R_2 per ottenere in uscita V_o = 20 V. Poiché dall'espressione precedente si ricava:

$$R_2 = R_1 \times \left(\frac{V_o}{V_{REF}} - 1\right)$$

Figura 4: *Stadio regolatore per una tensione di uscita variabile da 1,25 V a 20 V agendo sul potenziometro lineare R_2*

per R_2 si ottiene il valore:

$$R_2 = 240 \times \left(\frac{20}{1,25} - 1\right) = 3,6\ k\Omega$$

In pratica si porrà per R_2 un trimmer da 3,9 kΩ.
La figura 3 riporta la tensione di uscita ricavata in simulazione al computer tramite Spice. Il marker orizzontale *c* indica il valore della V_o. Si legge infatti *Yc* = 20,25 V.
Un'altra soluzione che meglio consente la variabilità della tensione di uscita e l'impostazione della massima V_o, è riportata nella figura 4 dove alla resistenza R_2 realizzata con un potenziometro lineare da 5 kΩ, si è posto in parallelo un trimmer da 22 kΩ. Si agirà su questo per fissare la V_{omax}, mentre si agirà sul potenziometro lineare per avere l'escursione della V_o da 1,25 V a 20 V. La figura 5 mostra, a questo proposito, le tensioni V_{omax} e V_{omin} ricavate in simulazione. Si legge infatti, *Yc* = 20 V e *Yd* = 1,25 V.

STABILIZZATORE CON V_o VARIABILE DA 0 A 30 V

Un'applicazione interessante dell'LM117 è proposta nella figura 6. Riprendendo l'espressione su riportata che fornisce la

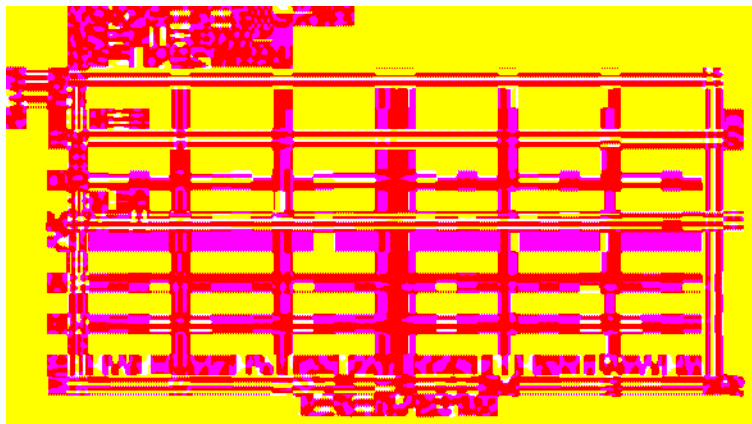
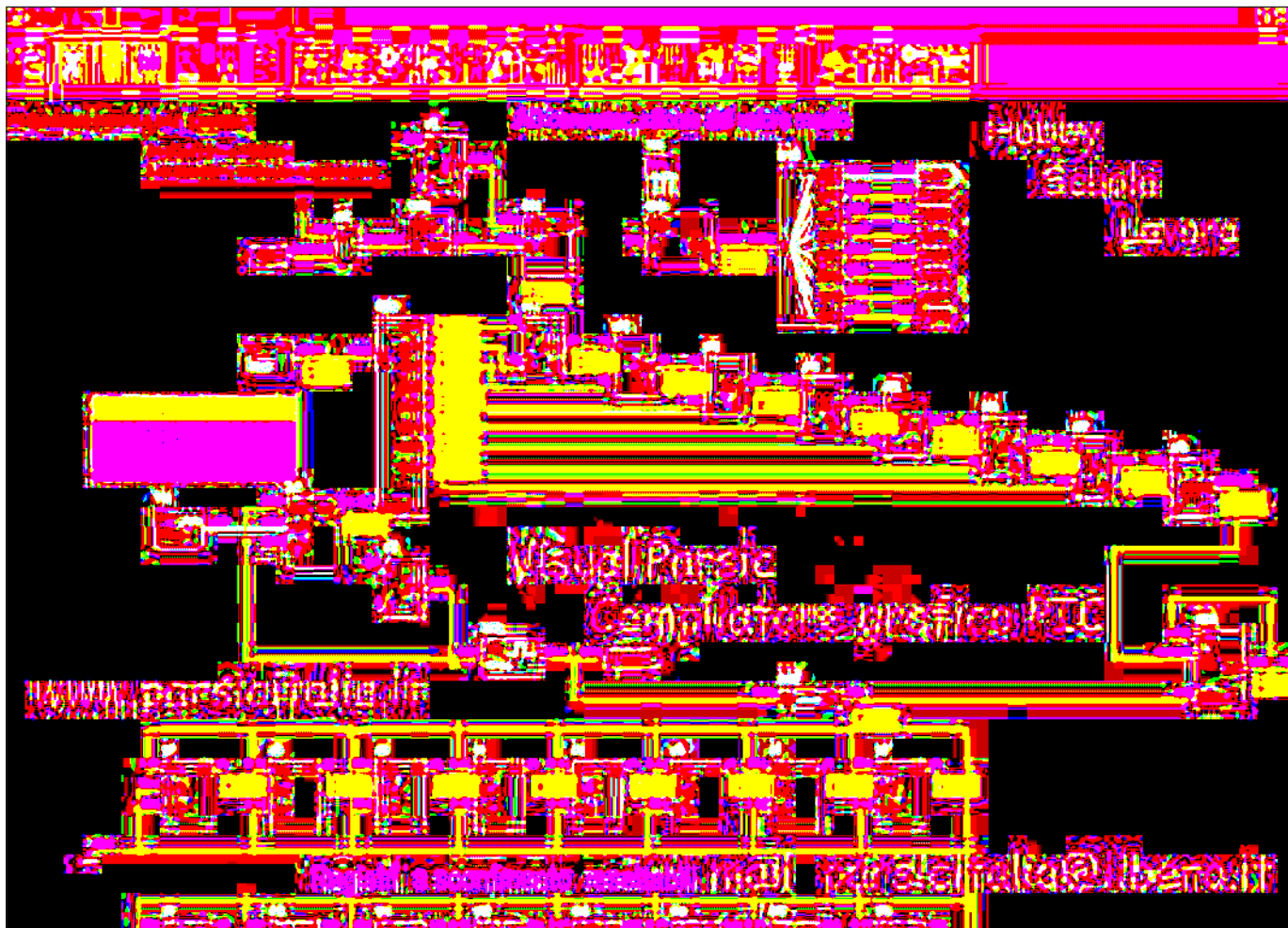



Figura 5: *I marker orizzontali c e d indicano rispettivamente, la tensione massima e minima in uscita nel regolatore di cui alla figura 4*

tensione V_o di uscita, si vede come sia possibile ottenere una V_{omin} = 0 solo a condizione di annullare anche il termine V_{REF}.
Per giungere a questo risultato si può fare, appunto, ricorso alla configurazione circuitale di cui alla figura 6. Qui il diodo zener D_z con V_z = 6,8 V, è polarizzato attraverso la resistenza R_4 da una tensione negativa in grado di portare a – 1,25 V il pin ADJ dell'IC. Il riferimento fuori massa, che porta sul pin ADJ una tensione eguale in valore (in modulo) ma di segno opposto alla V_{REF}, consente così di ottenere una V_{omin} pari a zero.
La R_3 è un trimmer, possibilmente multigiri, che consente l'azzeramento della tensione di uscita, mentre R_2 è sempre un potenziometro lineare che permette un'escursione della stessa tensione da zero a 30 V. Il diodo D_1 è utile per la protezione dell'integrato nel caso si manifesti un cortocircuito in uscita. In questa eventualità infatti il diodo costituisce la via di scarica per la corrente del condensatore C_3 la cui funzione è sia di migliorare la risposta ai transitori sia di contrasto all'innesco di oscillazioni.
Analoga funzione svolge il diodo D_2 nel caso si verifichi un cortocircuito in ingresso. D_2, bypassando l'LM117, costituisce infatti la via di scarica per il condensatore C_2. In assenza di D_2, infatti, la corrente di scarica entrerebbe nel regolatore attraverso il pin out e potrebbe danneggiare l'integrato.
La corrente di scarica dipende in tal caso da vari fattori: dal valore della tensione di uscita, dal valore attribuito alla capacità e dalla velocità di decremento della V_i.
In effetti, per capacità inferiori a 15 µF o 20 µF, l'LM117 è in grado di sostenere correnti di scarica molto elevate (anche dell'ordine dei 15 A), ma ove si realizzi un regolatore a tensione di uscita variabile con un qualsiasi IC a tre terminali, diverso dall'LM117, la corrente di scarica, anche se limitata, può portare a distruzione l'integrato. Per D_1 e D_2 si farà ricorso, nella configurazione di cui alla figura 6, a diodi del tipo 2N4002.

REGOLATORE SLOW TURN-ON

La flessibilità dell'LM117 è tale da trovare utilizzazione in un'infinità di circuiti. La figura 7, per esempio, propone uno stadio regolatore la cui tensione di uscita perviene al suo valore sono dopo un intervallo di tempo essenzialmente determinato dalla capacità C_t. La figura 8 mostra infatti come la V_o di uscita giunga al suo valore di 15 V solo dopo circa 5 s indicati dalla posizione del marker verticale *a*. L'intervallo (0 ÷ 5) s si potrà ovviamente espandere o contrarre rispettivamente aumentando o diminuendo il valore del condensatore C_t.

Un sistema del genere può, per esempio, essere molto utile negli impianti di amplificazione onde evitare nell'istante di accensione, la classica sollecitazione dei coni degli altoparlanti. Molti amplificatori (non necessariamente *professional*) sono infatti dotati di un'alimentazione ritardata che spesso fa riferimento al circuito qui esposto.

Figura 6: *Stadio regolatore con tensione di uscita variabile da 0 V a 30 V. Il diodo zener D_z porta sul pin ADJ dell'IC una tensione pari a circa –1,25 V, ossia una tensione eguale in valore ma di segno opposto alla tensione V_{REF} di riferimento dell'LM117. Ciò al fine di ottenere una tensione V_o che parta proprio da zero*

Figura 7: *Circuito per alimentazione ritardata. La tensione di uscita si porta al suo valore nominale dopo un intervallo di tempo imposto dal valore della capacità C_t*

STABILIZZATORE PER TENSIONI DUALI DA 1,2 A 20 V

L'IC LM117 trova il suo complementare nel regolatore per tensioni negative che ha sigla LM137 (o LM337). Utilizzando un LM117 e un LM137 si può facilmente realizzare un alimentatore per tensioni duali ossia per tensioni sia positive che negative rispetto al livello di massa.

Lo schema pratico di questa applicazione è proposto nella figura 9. Agendo sui trimmer da 2 kΩ si possono variare entrambe le tensioni di uscita da 1,2 V a 20 V applicando in ingresso le tensioni V_{i1} = + 25 V e V_{i2} = –25 V. Si fa presente che questo alimentatore è di tipo duale e non tracking dal momento che si possono ottenere tensioni positive e negative anche diverse una dall'altra. Nei ragolatori tracking, invece, agendo su un solo potenziometro si portano sempre al medesimo valore le tensioni di uscita.

Si rammenta, infine, che per tensioni di uscita elevate, ossia maggiori di 30 V, è opportuno ricorrere ai modelli previsti per alte tensioni che sono siglati LM117HV e LM137HV rispettivamente per V_o positive e negative.

NOTA PER IL LETTORE

La simulazione al computer dei circuiti qui esposti è stata eseguita col programma SPICE (*Simulation Program with Integrated Circuit Emphasis*) della Microcode Engineering Inc. (CircuitMaker).

Figura 8: *Andamento della tensione di uscita nel regolatore a ritardo di cui alla figura 7. Si noti come la tensione di uscita pervenga al suo valore finale (15 V) dopo circa 5 s indicati dal marker verticale a*

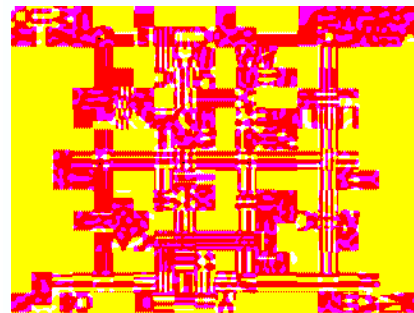
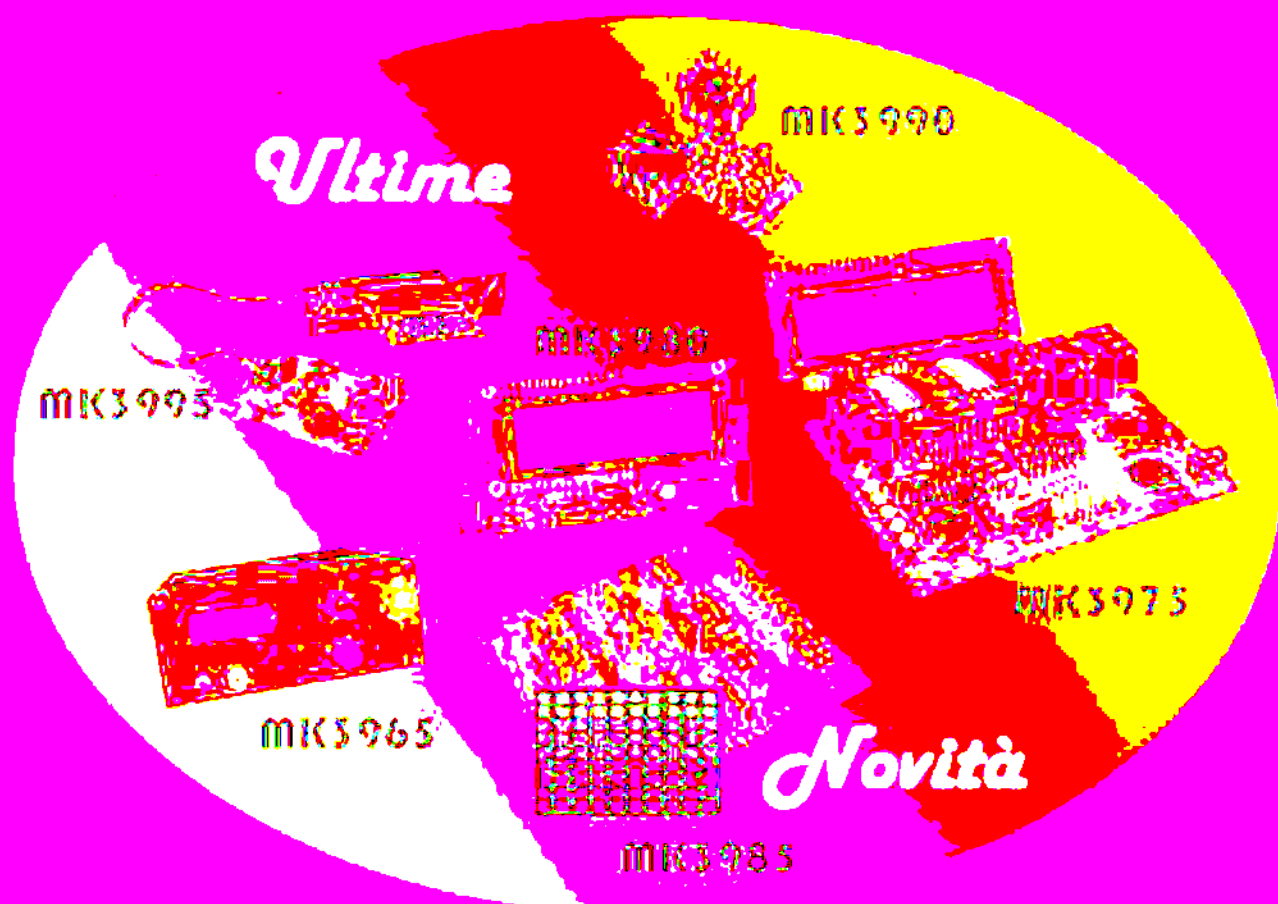



Figura 9: *Stadio regolatore di tipo duale che consente di ottenere tensioni positive e negative rispetto a massa*

VITAMINA C
undicesima parte

LE FUNZIONI DI LIBRERIA

di Antonio Di Stefano
a.distefano@farelettronica.com

In questa puntata ci occuperemo di un aspetto del linguaggio C abbastanza particolare: le funzioni di libreria. Queste funzioni permettono di aggiungere al linguaggio alcune importanti funzionalità, e consentono di gestire in modo semplice la memoria, i files, le funzioni matematiche e tanto altro. Vedremo con alcuni piccoli esempi l'uso delle più importanti funzioni e come queste possono semplificare molto lo sviluppo del codice.

Molte funzioni utili dell'ANSI C non sono parte integrante del linguaggio, ma sono disponibili come librerie che vanno richiamate nel proprio programma utilizzando la direttiva *#include*. Il motivo di questa scelta risiede nel fatto che queste funzioni sono in genere profondamente legate all'hardware della macchina su cui si vuole fare funzionare il programma. Il fatto di separare queste funzioni dal linguaggio vero e proprio permette di rendere i programmi in gran parte indipendenti dall'hardware, favorendo quindi la portabilità del codice. Non solo, utilizzando lo stesso meccanismo è possibile aggiungere anche funzioni personalizzate o rimpiazzare quelle esistenti per ottimizzare ancora di più le prestazioni in caso di necessità. Di seguito daremo uno sguardo alle classi di funzioni standard più importanti e alle librerie che le contengono.

FUNZIONI DI INPUT E OUTPUT

Abbiamo già visto nelle scorse puntate che l'ANSI C non dispone di istruzioni per comunicare con il mondo esterno, perfino la banalissima (si fa per dire) funzione *printf* deve essere richiamata da una libreria! La libreria in questione, che abbiamo richiamato con un **#include** in tutti i programmi visti fino ad ora, si chiama **stdio.h** ("standard I/O"), e come suggerisce il nome contiene molte funzioni utili per comunicare e scambiare dati con il mondo esterno. A volere essere precisi, i files con estensione .h che si richiamano dal programma non sono le librerie vere e proprie, ma dei file chiamati "header" che contengono la dichiarazione delle funzioni messe a disposizione dalla libreria, e le costanti. Le librerie vere e proprie sono disponibili di solito sotto forma di codice già compilato, pronto per essere inglobato ("linkato") nei nostri programmi.

Le funzioni messe a disposizione dalla libreria stdio.h permettono di comunicare con i dispositivi di *standard input* e *standard output* (dispositivi a caratteri, di solito lo schermo e la tastiera sui PC, o una porta seriale sui sistemi embedded), e di gestire i files.

La funzione in assoluto più utilizzata della libreria è la **printf**, che come abbiamo visto nelle scorse puntate serve per visualizzare del testo sullo standard output (di solito lo schermo), dando anche la possibilità di formattarlo e di visualizzare il contenuto delle variabili.

La funzione complementare di *printf* è la **scanf**. Questa funzione permette di leggere dei dati dallo standard input (di solito la tastiera), e di assegnarle

a delle variabili. La sintassi è la seguente:

```
int scanf("stringa di formattazione",
elenco variabili…);
```

La funzione permette anche di "de-formattare" il testo letto e di ripartirlo nella lista di variabili indicata. Un esempio di utilizzo è il seguente:

```
#include <stdio.h>
[…]
/* dichiarazione varviabili */
char testo[32];
int num;

/* richiede dati */
scanf("%32s %d", stringa, &num);
/* visualizza dati */
printf("\nStringa = %s", testo);
printf("\nIntero = %d", num);
[…]
```

Ci sono un paio di cose da notare. Innanzi tutto, a differenza delle funzioni di input di altri linguaggi, la scanf non permette di visualizzare un testo con la richiesta di immissione di dati, attende solamente che i dati indicati nella stringa di formattazione vengano immessi (si può ovviare con una printf prima). Altra cosa molto importante: la funzione non prende come argomento le variabili, ma il loro indirizzo! Occorre cioè passare le variabili per riferimento. Questo si fa utilizzando il carattere "&" nel caso variabili singole, oppure nel caso di vettori indicando il nome dell'array (che in realtà, come vedremo in seguito è un puntatore!).
Così come la funzione printf, la scanf ha un'infinità di caratteristiche e funzionalità aggiuntive, che per motivi di spazio non è possibile analizzare qui (rimando a qualche trattazione più completa o alle guide dei compilatori).
Altre due funzioni simili a printf e scanf, ma più semplici da utilizzare, sono **getc** e **putc**. La sintassi di queste funzioni è la seguente:

```
int getc(FILE *stream);
int putc(int c, FILE *stream);
```

Queste funzioni permettono di inviare o leggere singoli caratteri (nonostante il tipo utilizzato sia un intero), sia da files che dai dispositivi standard I/O. Questo le rende particolarmente adatte nel caso in cui lo standard I/O è rappresentato da una porta di comunicazione seriale per esempio. Un esempio di utilizzo è il seguente:

```
#include <stdio.h>
[…]
/* dichiarazione variabili */
int c;

/* ciclo infinito */
for(;;) {
/* legge carattere */
c = getc(stdin);
/* scrive carattere */
if (c!=EOF)
   putc(c, stdout);
}
[…]
```

Questo piccolo codice non fa altro che leggere un carattere dallo standard input e scriverlo sullo standard output, nel caso in cui questi dispositivi siano delle periferiche seriali in pratica implementa la funzione di "echo" (cioè rimanda indietro il carattere inviato per controllare che sia stato ricevuto correttamente). Da notare che le funzioni richiedono di specificare uno "stream" per i dati. Noi abbiamo specificato quelli di standard I/O, utilizzando due costanti contenute sempre nella libreria stdio.h: **stdin** ed **stdout**. Altre costanti importanti definite nella libreria sono **NULL** ed **EOF**, che specificano dei valori particolari associati ad esempio a puntatori vuoti o al carattere di fine file ("End Of File"). Da notare che la funzione getc restituisce proprio quest'ultimo carattere se lo stream è vuoto o se si è verificato un errore (da qui l'*if* che compare nel codice sopra).

Diamo adesso un'occhiata alle funzioni relative alla gestione dei files. Per il C un file è uno stream di dati, così come gli standard I/O. Per utilizzare i files è necessario prima di tutto dichiarare un apposito puntatore al tipo **FILE** (definito nella stdio.h). È necessario quindi aprire il file con la funzione **fopen**, eseguire le operazioni di lettura o scrittura, ed alla fine chiudere il file con **fclose**.

Le funzioni che si possono utilizzare per la lettura e la scrittura sono veramente tante, tra queste vale la pena citare le già viste **getc** e **putc** (e le varianti **fgetc** e **fputc**), usate per leggere e scrivere singoli caratteri (cioè singoli byte), e le funzioni **fprintf** ed **fscanf**, che funzionano come printf e scanf, ma operano su file di testo. La sintassi di queste funzioni è la seguente:

```
FILE *fopen("nome file", "stringa
modo");
int fprintf(FILE *stream, "stringa for-
mato", elenco variabili…);
int fscanf(FILE *stream, "stringa forma-
to", elenco variabili…);
int fclose(FILE *stream);
```

La funzione fopen prende come argomenti due stringhe, la prima indica il nome del file da aprire (eventualmente comprensivo di percorso), la seconda specifica la modalità di apertura del file. Un file può essere aperto in modalità lettura (stringa "r"), scrittura (stringa "w"), binario in lettura o scrittura ("rb" e "wb"), o in modalità "append" ("a"). In caso si successo la funzione restituisce il puntatore al file aperto, o in caso di errore o insuccesso NULL.
Le funzioni fprintf ed fscanf funzionano come le loro analoghe senza la "f" davanti, richiedono solamente in più il puntatore al file aperto che si vuole leggere o scrivere. Entrambe scrivono o leggono dal file una riga di testo alla volta. Una particolarità di tutte queste funzioni è che la lettura e la scrittura in un file avvengono in modo sequenziale, cioè ogni volta che si utilizza una funzione il puntatore al file viene incrementato, e l'operazione viene eseguita sul carattere o riga successiva. Per riportare il puntatore all'inizio del file si può utilizzare la funzione **rewind** [sintassi: **void** rewind(FILE *stream)].
Vediamo un piccolo esempio, che può risultare anche utile in pratica: il programma che segue legge un file di testo in formato UNIX/Linux (carattere ASCII di fine linea 0x0A), lo visualizza a schermo, e lo salva in formato Dos/Windows (caratteri di fine linea 0x0D 0x0A).

```
#include <stdio.h>

main() {
char *nomefine_in =
"Percorso/nome1.txt";
char *nomefine_out =
"Percorso/nome2.txt";
int c;
FILE *finein, *fileout;
int i;

/* apertura files */
filein=fopen(nomefile_in, "r");
fileout=fopen(nomefile_out, "w");

while((c=getc(filein))!=EOF) {

/* stampa a schermo */
printf("%c", c);

/* controlla tipo di carattere */
if (c!=0x0A) {
   /* testo normale */
   putc(c, fileout);
}
else {
   /* fine linea */
   putc(0x0D, fileout);
   putc(0x0A, fileout);
}
}

/* chiusura files */
fclose(filein);
fclose(fout);

}
```

Il programma apre i due file (in modalità lettura e scrittura rispettivamente, quello di scrittura viene creato automaticamente se non esiste già), ed esegue un loop per leggere tutti i caratteri, fino a quando non viene restituito il carattere di fine file EOF (notate una particolarità: la lettura e l'assegnazione fatta direttamente dentro la condizione del while). Il carattere letto viene controllato per capire se si tratta di un carattere normale o di un "a capo", nel primo caso esso viene copiato sul file di destinazione, altrimenti viene rimpiazzato con i due previsti dal formato Dos/Windows. Terminato il loop vengono chiusi i files prima di uscire dal programma.

FUNZIONI MATEMATICHE

Capita spesso di dovere utilizzare nei propri programmi funzioni matematiche un po' più complesse rispetto alle normali operazioni aritmetiche native del linguaggio. Per fortuna l'ANSI C mette a disposizione una libreria in cui sono state definite tutte le più comuni funzioni matematiche complesse: la libreria math.h. Le funzioni contenute nella libreria sono riportate nella tabella 1, notiamo quelle relative alle funzioni trigonometriche dirette, inverse ed iperboliche, quelle relative ai logaritmi, esponenziali, potenze, radici quadrate, ed ai valori assoluti. Come si può vedere, la maggior parte di queste funzioni utilizza variabili di tipo floating point (float, double, long double), questo implica che se si utilizzano tipi interi come argomenti o come risultati si ha una conversione (casting) automatica in ingresso o in uscita, che può comportare una perdita di precisione notevole! Inoltre è probabile che su sistemi più piccoli (a microcontrollore), non tutte le funzioni siano supportare, oppure sono supportare solo con tipi interi e con un fissato numero di bit. Per questo motivo prima di utilizzarle è bene leggere la documentazione del compilatore che si sta utilizzando.

LIBRERIA STDLIB.H

Una libreria che contiene molte funzioni importanti (in certi casi fondamentali!) è la **stdlib.h**. Una prima classe di funzioni contenute in questa libreria riguarda le operazioni su numeri interi (che come abbiamo visto non sono ben supportate dalla libreria math.h), un'altra classe riguarda la generazione dei numeri casuali, e un'altra ancora l'allocazione dina-

Funzione	Sintassi	Descrizione
abs	`int abs(int x);`	Valore assoluto di X
acos	`double acos(double x);`	Arco-coseno di X
asin	`double asin(double x);`	Arco-seno di X
atan	`double atan(double x);`	Arcotangente di X
atan2	`double atan2(double y, double x);`	Arcotangente di y/x
atof	`double atof(char *s);`	Converte la stringa s in float
ceil	`double ceil(double x);`	Arrotonda all'intero (> X)
cos	`double cos(double x);`	Coseno di X
cosh	`double cosh(double x);`	Coseno iperbolico di X
exp	`double exp(double x);`	Esponenziale e^X
fabs	`double fabs(double x);`	Valore assoluto di X (float)
floor	`double floor(double x);`	Arrotonda all'intero (< X)
fmod	`double fmod(double x, double y);`	X modulo Y (float)
log	`double log(double x);`	Logaritmo naturale di X
log10	`double log10(double x);`	Logaritmo decimale di X
pow	`double pow(double x, double y);`	X elevato a Y
sin	`double sin(double x);`	Seno di X
sinh	`double sinh(double x);`	Seno iperbolico di X
sqrt	`double sqrt(double x);`	Radice quadrata di X
tan	`double tan(double x);`	Tangente di X
tanh	`double tan(double x);`	Tangente iperbolica di X

Tabella 1: *Funzioni contenute nella libreria math.h*

mica della memoria e le funzioni legate al sistema.
Iniziamo con il primo gruppo, tra cui troviamo le utilissime funzioni per convertire le stringhe in numeri. Queste funzioni sono **atoi** ed **atol** (converte una stringa in un intero rispettivamente normale ed a doppia precisione). L'equivalente per numeri floating point è **atof** della libreria math.h. La funzione opposta si ottiene direttamente con printf o una delle sue varianti.
Il formato di queste funzioni è il seguente:

```
int atoi(char *s);
long atol(char *s);
```

Le funzioni che riguardano la generazione di numeri (pseudo)casuali sono **rand** ed **srand**. La prima restituisce un numero intero generato casualmente (in genere con distribuzione di probabilità uniforme), nel range da 0 a **RAND_MAX** (costante definita nella libreria). La seconda funzione invece serve per inizializzare il generatore di numeri casuali con un certo "seme". Questa possibilità è utile quando si vuole utilizzare una sequenza pseudocasuale ripetibile, o quando al contrario si vuole evitarlo. Per ricondurre il numero casuale ottenuto in un particolare range di valori è possibile utilizzare l'operatore modulo "%" (che da il resto di una divisione intera), o gli operatori binari. Vediamo un piccolo esempio:

```
#include <stdio.h>
#include <stdlib.h>

main() {
int i;

/* inizializzazione sequenza */
srand(5);

printf("\nSequanza di lanci di un
dado:");
for(i=0; i<10; i++)
   printf("\n%d", (rand()  % 6 + 1));
}
}
```

Questo programmino calcola 10 numeri casuali compresi tra 1 e 6, imitando il lancio di un dado. Dal momento che il generatore viene inizializzato sempre con lo stesso seme all'avvio, la sequenza sarà sempre uguale. Per ottenere un numero da 1 a 6 è stato sufficiente usare l'operatore modulo, ottenendo così un numero compreso tra 0 e 5, a cui è stato sommato 1.

Le altre importantissime funzioni presenti nella libreria sono quelle relative all'allocazione di manica della memoria. Tra queste meritano un cenno la **malloc** e la **free**. Grazie a queste funzioni è possibile richiedere una certa quantità di memoria (espressa in byte) per memorizzare dei dati, e poi rilasciarla quando non serve più. I possibili usi sono veramente tanti, però richiedono delle conoscenze un po' più approfondite per essere compresi bene.
Tuttavia vale la pena mostrare un esempio molto interessante. Abbiamo visto che quando si dichiara un array si stabilisce una volta per tutte la sua dimensione. Se si ha a che fare con dei dati di cui non si può conoscere l'esatta dimensione a priori questo può essere un problema, che può essere risolto banalmente sovradimensionando l'array. Esiste un modo molto più elegante ed efficiente: allocare l'array solo quando se ne conosce l'esatta dimensione!
Questa tecnica è mostrata nell'esempio seguente:

```
#include <stdio.h>
#include <stdlib.h>

main() {
int i, dim;
int *blocco_mem;

/* richiesta all'untente del numero
   di elementi da inserire */
printf("Numero elementi?\n");
scanf("%d", &dim);

/* allocazione dell'array */
blocco_mem = malloc(dim * sizeof(int));
/* allocazione non riuscita? */
if (blocco_mem == NULL) {
   printf("\nErrore di allocazione!!!");
   exit(EXIT_FAILURE);
}
```

```
/* immissione dei dati */
for(i=0; i<dim; i++) {
    printf("Inserisci numero?\n");
    scanf("%d", blocco_mem + i);
}

/* visualizzazione dei dati */
printf("Dati immessi: numero?\n");
for(i=0; i<dim; i++)
    printf("%d\n", blocco_mem[i]);

/* rilascia la memoria allocata */
free(blocco_mem);
}
```

Il programma chiede all'utente quanti dati vuole inserire, ed alloca un array su misura con la funzione malloc. Se l'allocazione non dovesse riuscire (per mancanza di memoria o altro) il programma termina, per fare questo viene utilizzata un'altra funzione della stdlib.h: **exit.** La costante EXIT_FAILURE è definita nella libreria assieme a EXIT_SUCCESS, e serve per indicare al sistema operativo, o ad un processo padre, se il programma è terminato in maniera normale o a causa di un errore. Da notare che la variabile per il blocco di memoria è stata dichiarata come un puntatore ad interi, a cui poi è stato assegnato l'indirizzo del blocco allocato restituito dalla malloc. Questo ci fa vedere che i puntatori possono essere poi utilizzati come dei normali array!

CONCLUSIONI

In questa puntata oltre alle più comuni funzioni di libreria sono stati introdotti alcuni concetti abbastanza avanzati, come l'uso dei puntatori o l'allocazione dinamica della memoria, che possono risultare molto utili quando si sviluppano programmi mediamente complessi. Nella prossima puntata ci soffermeremo proprio su questi aspetti, analizzandone più in dettaglio il funzionamento ed l'utilizzo.

FEBOT
prima parte

SPECIFICHE DEL PROGETTO

di Maurizio Del Corso *(m.delcorso@farelettronica.com)*
e Tiziano Galizia *(t.galizia@farelettronica.com)*

Chi ha seguito i forum sul sito di Fare Elettronica attendeva questo momento con grande ansia. Finalmente ci siamo: parte ufficialmente il progetto "FEbot", un robot interamente costruito da voi lettori. Un progetto allo stesso tempo ambizioso ed innovativo che ci accompagnerà per molti numeri a venire, mettendo alla prova le capacità tecnico/pratiche di chiunque intenda prenderne parte. Ma vediamo di cosa si tratta…

TANTA VOGLIA DI FARE

Non possiamo negarlo, la Redazione di Fare Elettronica ce la sta mettendo tutta per dare ai suoi lettori una rivista sempre aggiornata, ricca di progetti, di notizie ed articoli interessanti. Tramite i forum sul sito www.farelettronica.com e la rubrica Mailbox, tentiamo di rispondere alle vostre domande che ci arrivano copiosamente. Proprio dalle vostre domande, critiche e suggerimenti, abbiamo capito che i lettori di Fare Elettronica hanno raggiunto un buon livello tecnico ed hanno manifestato una grande passione per l'elettronica nonché tantissima voglia di impegnarsi in qualcosa di veramente interessante.

Perché allora non prendersi l'impegno di coordinare questo grande e volenteroso pubblico per costruire qualcosa che sia un punto comune a tutti, un pretesto per conoscersi, per scambiarsi informazioni e suggerimenti? Lo scopo non è soltanto quello di ottenere qualcosa di altamente professionale e sofisticato, ma quello di sentirsi parte di una squadra i cui componenti saranno sparsi per l'Italia e che ognuno di loro si impegnerà per raggiungere l'obiettivo comune: **la costruzione di un robot**.

Non sarà un'impresa facile, ma pensiamo che ne valga la pena e contiamo sulla vostra presenza, sui vostri suggerimenti e sulle vostre realizzazioni per gestire al meglio questo ambizioso progetto.

IL PROGETTO

OK, veniamo al sodo. FEbot è un robot di tipo modulare, cioè costituito da diverse schede elettroniche, collegate e comunicanti tra loro, ognuna delle quali svolgerà una funzione ben precisa, come ad esempio: il controllo motore, oppure la gestione di sensori di prossimità o la gestione di bracci meccanici e chi più ne ha, più ne metta! Qui entra in gioco la vostra fantasia:

potreste ad esempio progettare la scheda con i sensori di prossimità basata su infrarossi che potrà essere intercambiata con una equivalente agli ultrasuoni, progettata dal vostro migliore amico o da un lettore dalla parte opposta d'Italia. Ovviamente tutte le schede dovranno essere progettate secondo opportuni criteri che spiegheremo più avanti e ciascuna scheda sarà contrassegnata da un indirizzo assegnato da Fare Elettronica in fase di convalida.
L'unica cosa necessaria sarà una sorta di "mainboard", una scheda principale (della quale pubblicheremo il progetto nel prossimo numero) che abbia la funzione di coordinare tutte le periferiche connesse. La comunicazione tra le diverse schede avverrà secondo un protocollo di comunicazione seriale sincrono del quale forniremo le specifiche in modo da definire un "linguaggio universale" per tutte i moduli progettati.

Come si partecipa

Se state leggendo queste righe, significa che in qualche modo l'idea vi stuzzica quindi, prima di addentrarci nelle questioni tecniche, illustriamo in che modo si può partecipare all'iniziativa.
Sul sito Internet di Fare Elettronica troverete una sezione interamente dedicata a questo progetto ed il primo passo è quello di comunicare la vostra partecipazione registrandovi alla pagina www.farelettronica.com/febot indicando quale è la funzione che intendete assolvere con la scheda da voi progettata.
La partecipazione è aperta anche ai gruppi ed in questo caso ogni partecipante dovrà registrarsi indicando lo pseudonimo del gruppo di appartenenza. La partecipazione è completamente gratuita e non comporta l'acquisto di alcun kit o altro materiale. L'unico costo, sostenibilissimo, è la vostra passione!

Ho realizzato un modulo: e ora?

Una volta che la scheda è stata realizzata e testata dovrà essere convalidata dalla Redazione di Fare Elettronica in modo che divenga ufficialmente parte del FEbot. La convalida avviene verificando i requisiti di progettazione hardware/software. Una volta convalidata le sarà assegnato un indirizzo (un byte) ed il progetto sarà pubblicato integralmente sul primo numero disponibile della rivista, completo dei nomi e foto degli autori nonché della lista dei comandi supportati con la relativa descrizione.

Contemporaneamente sarà pubblicata una descrizione sul sito Internet della scheda progettata illustrandone le caratteristiche tecniche senza dimenticare di valorizzare la bravura del suo autore!

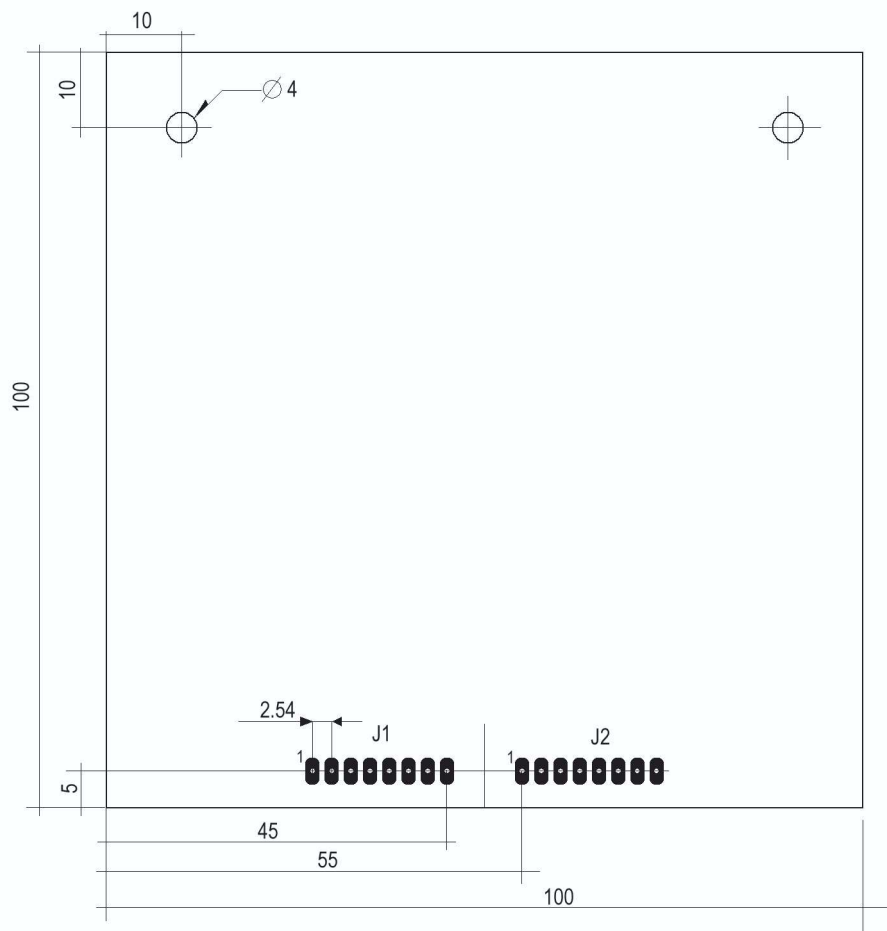



Figura 1: *Le dimensioni (in millimetri) e la struttura di una singola scheda*

La procedura sarà la seguente:

- Il partecipante (o il gruppo) una volta terminata la progettazione di un modulo, invia alla redazione tutta la documentazione relativa al modulo progettato (progetto, descrizione, foto, firmware, eccetera).
- La Redazione esamina il progetto e si accerta che risponda alle specifiche correnti.
- In caso di esito positivo, la Redazione assegna al modulo un "part number" ed un indirizzo, lo comunica al progettista il quale ha il compito di integrarlo nel proprio firmware secondo quanto richiesto dal protocollo di comunicazione seriale.
- Il progetto così completato viene interamente pubblicato sul primo numero disponibile della rivista (previa autorizzazione scritta dell'autore).
- La descrizione delle caratteristiche del nuovo modulo vengono inserite immediatamente nell'apposita sezione del sito e della rivista.

Uno stesso partecipante (o gruppo) può contribuire al progetto anche con più moduli, il limite è solo la fantasia di ciascuno di voi!

LE QUESTIONI TECNICHE

L'hardware

L'hardware è completamente libero: potete usare microcontrollori PIC o qualsiasi altro componente intelligente e tutto ciò che ritenete più idoneo al vostro progetto. L'importante è che il circuito stampato abbia la struttura di figura 1 in modo da assicurare la compatibilità con le schede progettate da altri partecipanti.

Il circuito stampato potrà essere a singola o doppia faccia, ma è fondamentale che sia dotato dei connettori J1 e J2 che sono a

	J1	
1	**D0**	Pin di I/O direttamente collegati al micro presente sulla scheda madre (di cui parleremo approfonditamente nel prossimo numero).
2	**D1**	
3	**D2**	
4	**D3**	
5	**D4**	
6	**D5**	
7	**D6**	
8	**D7**	
	J2	
1	**R/W**	Identifica la direzione del flusso dati da/verso il micro della scheda madre. Se questo pin è a livello alto, il flusso è verso la scheda madre (operazione di lettura).
2	**BUSY_P**	Se a livello alto notifica che è in corso una transazione di dati sui fili D0÷D7.
3	**S_DATA**	Il filo su cui transitano i dati della comunicazione seriale.
4	**GND**	Massa.
5	**GND**	Massa.
6	**CLK**	È il segnale di sincronizzazione per la comunicazione seriale.
7	**BUSY_S**	Se a livello alto indica che è in corso una comunicazione seriale.
8	**Vcc**	Positivo di alimentazione.

Tabella 1: *Descrizione dei connettori J1 e J2*

Figura 2: Struttura del pacchetto dati seriale

passo 2.54. Questi dovranno essere passanti, maschio sul lato superiore e femmina su quello inferiore. In questo modo le varie schede potranno essere connesse una sull'altra come un sandwitch. I fori sul lato opposto rispetto ai connettori, consentiranno il fissaggio delle diverse schede tramite opportuni distanziatori. I tre lati liberi della scheda possono essere utilizzati per connettori specifici (come ad esempio il collegamento con motori, luci, eccetera).

Il bus di comunicazione

Attraverso i connettori J1 e J2, le schede possono comunicare tra loro e con la scheda madre. I segnali presenti su J1 e J2 costituiscono un bus di comunicazione la cui struttura è riportata nella tabella 1.

Il protocollo di comunicazione seriale

La comunicazione seriale tra due moduli o tra un modulo e la scheda madre, avviene utilizzando un protocollo di comunicazione che prevede pacchetti dati di lunghezza fissa. Quando un modulo intende iniziare una comunicazione seriale deve seguire il seguente algoritmo:

- Analizza la linea **BUSY_S**: se è a livello alto attende altrimenti la pone a livello alto per iniziare la trasmissione.
- Trasmissione del pacchetto dati e del segnale di sincronismo.
- Alla fine della trasmissione riporta a zero la linea **BUSY_S**.

Il pacchetto dati che viene trasmesso è composto da 19 byte e la sua struttura è quella di figura 2:

- **INDIRIZZO**: è l'indirizzo del modulo al quale è destinato il pacchetto dati. L'indirizzo è espresso su un singolo byte e vale 01H (00000001) per la

Valore	Comando	Descrizione
FFH (11111111)	**Accensione**	All'arrivo di questo comando il modulo dovrà portarsi nella modalità operativa.
00H (00000000)	**Standby**	Questo comando pone in standby il modulo che lo riceve. Un modulo in standby non è operativo, ma è in grado di ricevere ed interpretare i comandi che transitano sul bus.
01H (00000001)	**Reset**	Provoca il riavvio del modulo.
02H (00000010)	**ID**	Alla ricezione di questo comando il modulo risponde inviando, nel campo dati, il proprio part number (assegnato da Fare Elettronica a seguito della procedura di convalida) formato da 6 caratteri seguito da altri 10 caratteri in cui il progettista del modulo può inserire dati a piacimento (il proprio nome, pseudonimo del gruppo, data, eccetera).
03H (00000011)	**LOW_BAT**	Questo comando è seguito da un dato che esprime il valore della tensione della batteria e viene inviato se la batteria scende al di sotto di un valore prefissato.
04H (00001000)	**PC_conn**	È un comando che informa i moduli che un PC è stato connesso alla porta seriale RS232 della scheda madre.

Tabella 2: Descrizione dei comandi standard

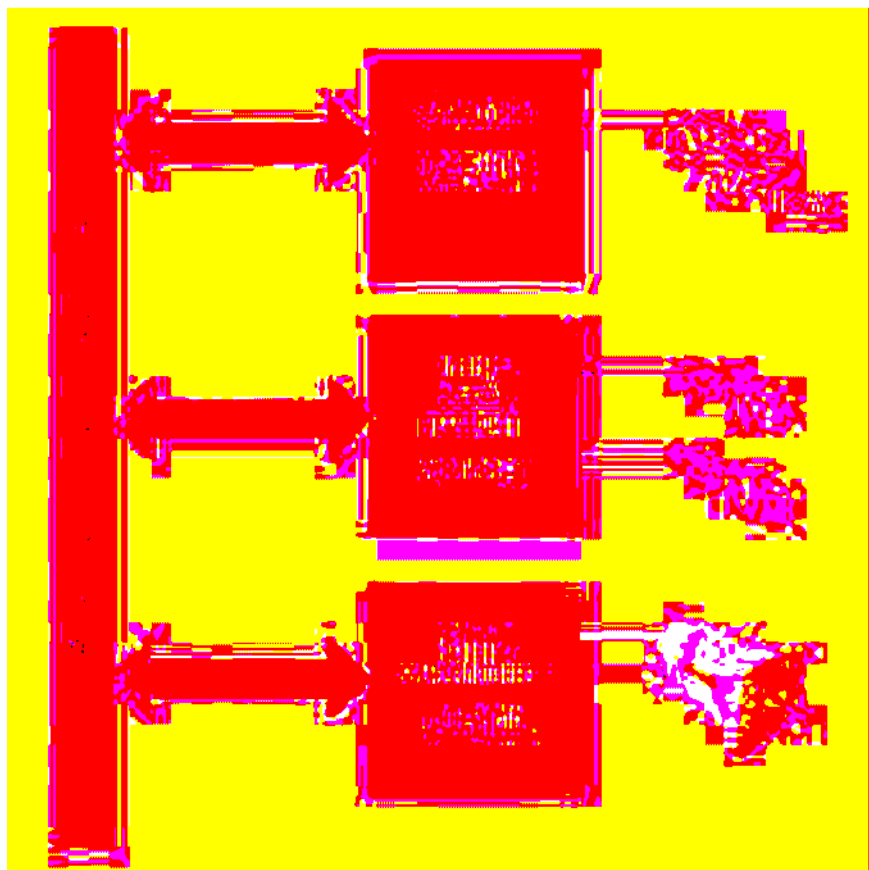



Figura 3: Una possibile configurazione elementare del FEbot

scheda madre; ciascun modulo risponde al proprio indirizzo e all'indirizzo broadcast FFH (11111111). L'indirizzo broadcast consente di inviare un pacchetto dati a tutti i moduli contemporaneamente.

- **COMANDO**: uno dei comandi riconosciuti dal modulo destinatario oppure uno dei comandi comuni a tutti i moduli.
- **DATI**: eventuali dati da trasmettere (questo campo non è significativo se il comando non comporta un trasferimento dati). Il pacchetto dati è sempre composto da 16 byte, i byte non utilizzati valgono 0 (00000000).
- **CRC**: campo di controllo sull'integrità dei dati trasmessi.

La comunicazione seriale deve avvenire a velocità non superiori ai 10 Kbps. Nei prossimi numeri verranno presentate alcune routine per PICmicro in grado di gestire questo protocollo di comunicazione.
Ciascun modulo realizzato, oltre a riconoscere una serie di comandi per la gestione delle funzionalità previste (questi comandi sono definiti da chi progetta il modulo), deve essere in grado di riconoscere i comandi standard riportati in tabella 2.

PER ESEMPIO...

In figura 3 è riportata una ipotetica struttura del FEbot che racchiude un po' tutte le caratteristiche viste fino ad ora.

In questo caso sono stati inseriti due moduli: un modulo sensori prossimità ed un modulo di controllo trazione. Il primo comunica con il secondo inviando, ad esempio, il comando di arresto motore se viene rilevato un ostacolo.
Poiché entrambi i moduli riconoscono i comandi standard, le scheda madre può decidere di spegnere il robot (quindi tutti i moduli) se il livello della batteria scende al di sotto del valore di guardia.
Ovviamente quello riportato è solo un esempio. Date sfogo alla vostra fantasia nell'inventare nuovi moduli e nuove funzionalità al FEbot. Ogni mese noi della Redazione vi forniremo idee e spunti, pubblicando anche i vostri suggerimenti.

IN CONCLUSIONE

Per questo mese è tutto. Non perdetevi l'evoluzione del FEbot che dipende solo dal vostro entusiasmo e dalla vostra passione. Noi cercheremo di aiutarvi in tutti i passaggi relativi alla progettazione, oppure nella soluzione di un problema che vi impedisce di proseguire nella realizzazione della vostra opera.
Vi ricordiamo che sul sito di Fare elettronica è disponibile un forum dedicato al FEbot, che potete utilizzare per confrontarvi con gli altri partecipanti, oppure per chiedere consigli o fare proposte.

Attendiamo impazienti le vostre idee, i vostri suggerimenti ed i vostri consigli e non perdete il prossimo numero con il progetto della scheda madre!

ASPIRONE: UN ROBOT ASPIRAPOLVERE

seconda parte

di Marco Fabbri
marnic@roboitalia.com

Riprendiamo come promesso l'argomento Aspirone, il nostro robot aspirapolvere. Nella precedente puntata ci siamo occupati della costruzione meccanica, analizzando solo marginalmente il cervello e i pensieri del nostro collaboratore domestico.

IL CERVELLO

Come accennato nella puntata precedente ho utilizzato una scheda commerciale denominata MarkIII, questa scheda viene proposta principalmente come "l'elettronica programmabile" per i robot da minisumo.
È una scheda basata sul PIC16F877, un microcontrollore programmabile prodotto dalla Microchip che ha caratteristiche decisamente interessanti per applicazioni hobbistiche e, non da sottovalutare, un costo più che abbordabile.
In realtà il circuito è molto flessibile e a mio avviso può costituire la base per molte realizzazioni, la scheda viene venduta in rete sia in Italia che all'estero, lascio a voi le valutazioni di convenienza fra i vari offerenti.
La scheda viene fornita "da montare" questo significa che riceverete il circuito stampato e i componenti da saldare.
Come potete vedere in figura 1 e figura 2 non ci sono componenti SMD e questo significa che con un normale saldatore (e un po' di esperienza) non dovreste avere problemi, in rete trovate anche l'elenco componenti e il layout dello stampato, le numerazioni corrispondono e non ci sono errori.

Per il montaggio dei componenti io inizio da quelli con profilo più basso, solitamente le resistenze, poi i diodi e così via.
Ricordate di fare attenzione ai componenti che hanno una polarità (diodi e condensatori elettrolitici) o un verso (integrati e zoccoli per gli stessi).
Se acquistate la versione base avete 2 integrati (oltre al PIC) che sono forniti senza zoccolo, per antiche abitudini io gli integrati li monto su zoccolo, sicuramente le spesa non è eccessiva e siete sicuri di non "cuocere" il circuito integrato durante la saldatura.
Ultimato il montaggio prendiamo in esame lo schema e le caratteristiche di questa scheda.

LO SCHEMA ELETTRICO

Come anticipato lo schema elettrico e il disegno del circuito stampato, lo potete scaricare gratuitamente dal sito del produttore della scheda:
http://www.junun.org/MarkIII/Info.jsp?item=3.
Noi ne esaminiamo un blocco alla volta.

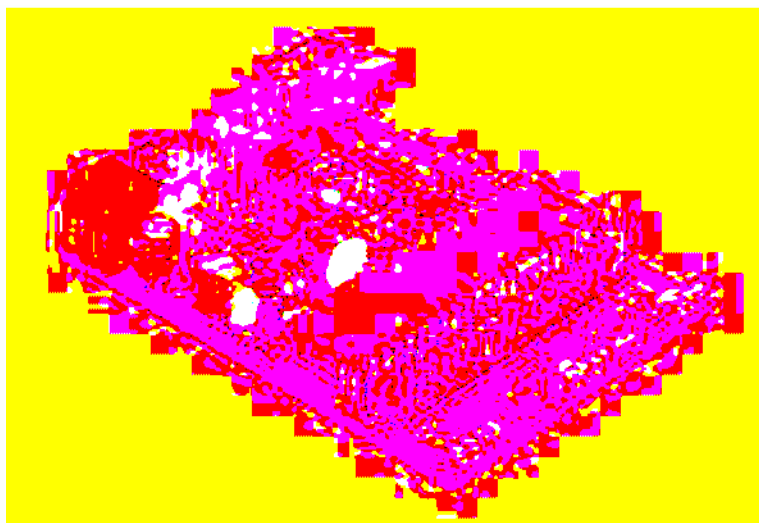



Figura 1: La scheda MARKIII

Partiamo ovviamente dallo stadio di alimentazione. Il circuito prevede una doppia alimentazione per dividere i motori dall'elettronica, questo serve a fare in modo che gli spunti dei motori non producano una caduta di tensione tale da resettare il PIC generando il riavvio del programma. Nel connettore T1 troviamo appunto 3 morsetti, il numero 1 è l'alimentazione dei motori (V+), il numero 3 per l'elettronica (Vin), il numero 2 è a massa comune alle due alimentazioni (VSS).

La tensione V+ va scelta ovviamente in base ai vostri motori,non esiste infatti nessuna regolazione o protezione su questa linea di alimentazione, se utilizzate dei comuni servocomandi questi sono normalmente alimentati a 6V ma anche a 7,2V spesso il costruttore indica i dati di coppia e velocità, io li alimento a 8V e non mi danno problemi.

Dalla parte Vin troviamo invece un diodo di protezione contro le inversioni di polarità e un regolatore della Maxim, precisamente un MAX667CPA che fornisce in uscita 5 V stabilizzati, un po' di condensatori a monte e a valle che filtrano i disturbi, un led verde che segnala la presenza di tensione ed infine un led rosso per la segnalazione di tensione insufficiente, in pratica, quando si accende il led rosso significa che il regolatore non è in grado di mantenere i 5 V e questo a causa probabilmente di una tensione a monte troppo bassa. La tensione così stabilizzata viene indicata nello schema come VDD e la ritroveremo nei successivi blocchi.

Il prossimo blocco che analizziamo è apparentemente molto semplice quello relativo al "line Follower" o inseguitore di linea. Il circuito realizza tre partitori di tensione attraverso R8, R9, e R10 e i relativi fototransistor. Le uscite dei partitori indicate con LineR, LineC e LineL vanno collegate a 3 porte del PIC e rispettivamente RE0, RE1, RE2. C'è poi una serie di tre diodi led usati per illuminare o meno a seconda della riflessione sulla linea da seguire i fototransistor, come vedete i led sono in serie per cui dovrete collegarli tutti anche se uno non vi dovesse servire. La cosa interessante è il collegamento del partitore al PIC, sono infatti utilizzate delle porte che accettano anche un segnale analogico, questo significa che potremo decidere via software se impostare l'ingresso come digitale e quindi avere delle reazioni tipo ON/OFF oppure impostarlo come analogico e modulare la reazione in funzione del valore letto.

Figura 2: I componenti necessari a realizzare la scheda MARKIII

Nel caso di Aspirone, oggetto di questa puntata, al posto dei fototransistor ci sono collegati dei normali finecorsa e quindi il funzionamento in digitale è più che sufficiente ma è una possibilità da tenere a mente per possibili sviluppi.

Cambiamo blocco e passiamo al OOPic Support. Nella versione base questa opzione non viene fornita, in pratica vi manca il chip 24LC32A, avete però il connettore J2 e JP1, il J2 è il connettore dove ci troviamo i segnali relativi al protocollo I2C, JP1 serve invece attraverso il posizionamento di un ponticello ad abilitare il protocollo a seconda della versione scelta del MarkIII, in particolare per la ver-

sione base non serve mettere nessun ponticello. Attraverso il protocollo I2C potremo comunicare con tutta una serie di dispositivi e/o memorie dove salvare/leggere dati o acquisire informazioni dal mondo esterno. Anche questa caratteristica ci offre molte possibilità di espansione e quantomeno la possibilità di lavorare di fantasia.
Saltiamo ora al blocco chiamato Serial Interface. Questa parte è costituita essenzialmente da un solo integrato, il DS232 che converte segnali con logica TTL (0-5V) in segnali compatibili con lo standard RS232 (-12 +12V). In questo modo si realizzano due funzioni importanti, una è quella di poter dialogare con la scheda attraverso la seriale con il nostro PC e quindi avere informazioni di vario tipo, dalla lettura dei sensori allo stato degli ingressi, la seconda, non meno importante, è quella di realizzare la famosa programmazione "in-circuit" ovvero senza smontare il PIC dal circuito (e senza aver bisogno di un programmatore). Per questa seconda funzione, il PIC che vi viene fornito non è vuoto ma contiene al suo interno una piccola routine chiamata Picloader. Spendiamoci due parole. Il Picloader per la sua posizione in memoria, viene eseguito ad ogni avvio/reset del PIC e non fa altro che monitorare per 5 secondi le porte relative alla comunicazione seriale, se non trova dati in arrivo, passa il controllo al programma principale, diversamente, si mette in attesa dei comandi relativi alla programmazione. È facile vedere all'opera il Picloader, collegando il PC alla scheda attraverso la seriale e avviando il terminale di Windows. In rete è facile trovare il Picloader e caricarlo sul proprio PIC qualora non ci sia, avrete però bisogno di un programmatore per completare l'operazione.
Andiamo avanti e parliamo dei connettori, J3 e J4 che sono relativi ai motori; sulla serigrafia dello stampato sono chiaramente indicati i colori dei fili secondo gli standard dei servocomandi, in pratica arrivano il più e il meno per l'alimentazione e il segnale PWM, inutile dire che utilizzando dei servocomandi ci semplifichiamo la vita. J5 e J6 sono i connettori previsti per i sensori GP2D12, sono sensori che misurano la distanza di un ostacolo, hanno un'uscita analogica e sono infatti collegati alle porte RA2 e RA3 settabili come ingressi analogici, anche qui nulla vieta di sostituirli con altri sensori (con uscita compatibile) per misurare altre grandezze. J7 è il classico connettori a 9 poli tipo RS232. J1 è "il" connettore, sono infatti riportati tutti i pin utili del PIC oltre alle tensioni di alimentazione per poter espandere senza vincoli la nostra scheda. Ovviamente nel farlo dovremo fare attenzione a cosa siamo già collegati sulla scheda ed eventualmente isolare il blocco che non ci interessa.
Arriviamo all'ultima parte, il blocco con il cervellone. In realtà non c'è molto da dire, oltre all'identificazione dei diversi pin e al loro collegamento, troviamo il quarzo per il clock e il pulsante di reset, ricordatevi che dopo il rilascio del reset avete 5 secondi dove non succede nulla per colpa del *Picloader*.

Figura 3: *Circuito di alimentazione da montare su scheda millefori*

Visto e compreso lo schema non dovrebbe essere difficile attivare Aspirone, abbiamo due servomotori e il collegamento è obbligato a J3 e J4, stessa cosa dicasi per i GP2D12.
Per i baffi, io ho usato tre finecorsa e in questo caso si può scegliere se collegarli uno ad uno ai 3 partitori resistivi del line follower (contatti 3 e 4) oppure metterli tutti in serie ed utilizzare un solo partitore riservandosi gli altri due per sviluppi futuri.
Per l'alimentazione ho realizzato il circuito che vedete in figura 3 su un bezzo di basetta millefiori e direi che si commenta da solo; come interruttore generale ho

utilizzato un deviatore e l'ho collegato in modo che, quando spengo Aspirone, posso collegare il carica batterie al connettore per la ricarica.

IL PENSIERO

Parliamo ovviamente del programma, ma visto che gli abbiamo dato un nome, costa poco immaginare di dare al nostro nuovo aiutante anche una personalità.

Il programma è scritto in *PicBasic*, successivamente compilato e quindi caricato nel microcontrollore.

Ovviamente si può fare la stessa cosa utilizzando altri linguaggi di programmazione.

Il programma, che potete scaricare dal sito Web di Fare Elettronica, utilizza un *interrupt* per mandare gli impulsi, attraverso il comando PULSOUT, alle porte dei servocomandi, questa tecnica rende facile inviare con continuità il segnale PWM ma limita un po' l'uso del comando PAUSE, per questo motivo quest'ultimo viene sostituito da due cicli FOR-NEXT annidati e sarà proprio sul numero di cicli che dovremo andare ad agire per ottenere le rotazioni giuste di 90 e 180 gradi.

L'utilizzo di due cicli annidati permette di avere due regolazioni, una grossolana e una fine, se aumenterete la variabile del ciclo esterno anche di una sola unità è come se raddoppiaste il numero della variabile interna.

Una volta caricato il programma è necessario eseguire un po' di tarature.

Per prima cosa dovrete verificare che il vostro robot in assenza di ostacoli vada diritto. Per fare questo appoggiatelo in terra lontano da pareti o ostacoli e avviatelo; a meno di una fortuna sfacciata il robot tenderà a girare a destra o a sinistra, regoleremo i trimmer dei servocomandi per correggere la rotta, con un paio di operazioni dovreste riuscire a centrare l'obiettivo.

La seconda taratura è la rotazione di 180 gradi. Avviate Aspirone e lasciatelo andare contro un ostacolo, quando i baffi urteranno il robot retrocederà di qualche centimetro e inizierà a ruotare su una sola ruota, se la rotazione è scarsa dovrete aumentare i cicli for-next di cui abbiamo parlato, al contrario se è eccessiva li diminuirete.

Potete a questo punto impostare i cicli per i 90 gradi a circa un quarto di quelli per i 180 (vengono usati entrambi i motori e quindi metà tempo per metà spazio), ricaricate il programma e lasciate andare Aspirone, dopo il primo contatto con un ostacolo avremo la rotazione di 180 gradi da un lato (a destra) seguite il robot con un foglio bianco che gli avvicinerete al lato sinistro poco prima del successivo impatto frontale, Aspirone se ne accorgerà e dopo l'impatto ruoterà solo di 90 gradi verso destra; se non lo farà modificate la variabile EYE_soglia diminuendone il valore così da rendere più sensibili gli occhi del vostro robot.

Vi spiego il perché di questo comportamento. Dopo la prima rotazione a destra doveva seguirne una a sinistra per continuare a traslare lungo la stanza, il muro sul lato però gli impediva di eseguire questa manovra, per cui per non ripassare sullo stesso punto il robot ha scelto di eseguire il cambio di direzione come abbiamo visto nella prima puntata di questo articolo. Questa scelta ha un particolare effetto collaterale, infatti sembra che, osservandolo al lavoro, prevalga un comportamento che predilige gli angoli e i bordi del muro (con grande soddisfazione per chi osserva).

Verificate le varie condizioni osservando per qualche tempo la vostra creazione al lavoro, le difficoltà maggiori io le ho trovate nei baffi, o meglio nel rischio che questi si incastrino da qualche parte, anche spargendo un po' di polvere in qualche punto difficile o strategico per vedere se il lavoro viene eseguito a dovere, diversamente… niente ricarica della batteria!

Naturalmente questo programma è solo una delle tante possibilità e sicuramente state già pensando ad una soluzione migliore, ma… anche io e le sorprese o gli upgrade non sono ancora finiti. Se avete notato la figura 9 della prima puntata, vi sarete accorti che sulla sinistra si nota un piccolo pulsante a linguetta che a quanto pare non ha alcuna funzione, così come il connettore a due pin posizionato sopra, curiosi?

Servono per la taratura della bussola elettronica con la quale, oltre ad imparare qualcosa sul protocollo I2C avremo un robot che sempre più prenderà coscienza del mondo esterno.

Non perdetevi i prossimi numeri.

TUTTO IL CORSO "PIC® MICROCONTROLLER BY EXAMPLE" SU CDROM

C'è chi si è perso qualche puntata, c'è chi non trova più proprio quel numero della rivista, c'è chi oramai è abituato a sfogliare libri in formato elettronico… Da oggi tutti questi problemi sono risolti perché l'oramai famoso corso PIC® Microcontroller By Example è ora interamente disponibile su CDROM. Un CD che raccoglie tutte le puntate in formato Acrobat PDF e moltissimi contenuti speciali. Scopriamolo insieme.

PIC® MICROCONTROLLER BY EXAMPLE

Era il 1998 quando Fare Elettronica pubblicò la prima puntata del corso che all'epoca era noto come "PIC By Example". Visto l'enorme successo e le grandi richieste da parte dei lettori, il corso è stato rivisto, corretto e aggiornato, quindi pubblicato nuovamente a partire dal numero doppio Ottobre/Novembre 2003 che è anche il primo numero di Fare Elettronica sotto la nuova casa editrice Inware Edizioni. L'interesse da parte dei lettori non accenna a diminuire, come testimoniano le domande ed i suggerimenti pervenuti copiosamente in Redazione. Per premiare questo interesse e questa inesorabile sete di sapere, Fare Elettronica ha raccolto tutta l'ultima versione del corso in un unico CD-ROM arricchito di moltissimi contenuti speciali.

NAVIGANDO IL CD-ROM

Il CD-ROM è predisposto per l'avvio automatico se questo è abilitato sul vostro PC. In alternativa, lanciando il programma start.exe ed accettando la nota sul copyright, si accede al menu di navigazione (figura 1) da cui è possibile esplorare tutti i contenuti. Da qui, cliccando sui titoli, si accede al file pdf di ciascuna delle otto parti del corso. I file pdf, ad alta risoluzione, sono leggibili con Acrobat Reader anch'esso incluso nel CD e possono essere stampati ottenendo risultati paragonabili a quelli tipografici. Cliccando su "Indice del contenuto" si accede all'indice degli argomenti trattati nella specifica parte del corso (figura 2). Cliccando invece su "Downloads" si accede ai file zip contenenti i pro-

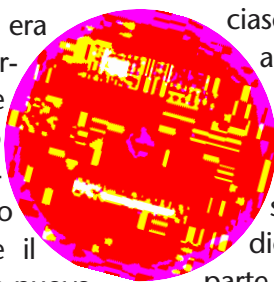



Figura 1: *Il menu principale*

Figura 2: *L'indice del contenuto della prima parte*

Figura 3: *La prima pagina del corso visualizzata da Acrobat*

grammi sorgenti utilizzati nelle diverse parti.

I CONTENUTI SPECIALI

Si accede ai contenuti speciali mediante l'apposito pulsante nel menu principale.

Datasheet

In questa sezione è possibile accedere al datasheet del PIC16F84A utilizzato nel corso, al datasheet del display LCD 16x2 e del sensore ad infrarossi utilizzati negli esempi applicativi.

Compilatori

La sezione compilatori comprende una gamma di compilatori per PICmicro® che consentono di programmare i dispositivi utilizzando linguaggi di alto livello come il C o il Basic. I compilatori proposti sono il PIC-C (versione Lite), il PicBasic Pro (compilatore Basic) ed il PCW, compilatore C della HiTech. Ove disponibile, il compilatore è corredato del relativo manuale utente.

MPLAB

MPLAB® è l'ambiente di sviluppo di Microchip™ per tutte le famiglie di PICmicro®. La versione demo comprende il compilatore assembler (utilizzabile per compilare i progetti presentati nel corso) e l'help in linea che ne descrive l'utilizzo.

CD FE2003

La versione demo del CD-ROM "Annata Fare Elettronica 2003". La versione demo comprende il pdf del solo numero 222 (Dicembre 2003) con i relativi files. Il numero 222 è interessante in quanto comprende il progetto di un programmatore per PICmicro® che può essere utilizzato per le esercitazioni proposte nel corso.

Software

Nella sezione software troviamo le versioni demo di **sPlan 5.0** per il disegno degli schemi elettrici, di **sPrint Layout 4.0** per il disegno dei circuiti stampati e di **Front Designer 2.0** per il disegno dei pannelli frontali.

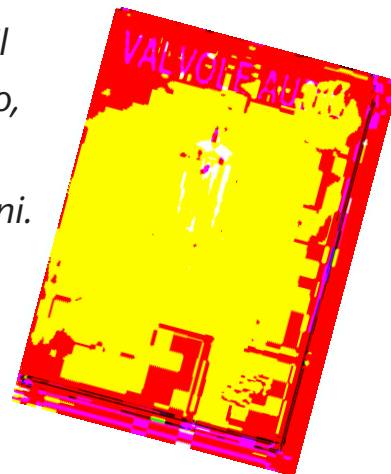
Figura 4: *I contenuti speciali*

Figura 5: *La demo del CD "Annata Fare Elettronica 2003"*

IL PREZZO

Il CDROM ha un prezzo speciale di € 15,00 e può essere ordinato telefonando allo 02-66504794 oppure scrivendo a **ordini@farelettronica.com**.

VALVOLE AUDIO

Pianeta Elettronica presenta "Valvole Audio" il nuovissimo CD-ROM sviluppato da Eurocom-Pro, azienda dedicata allo sviluppo di software e multimedia per elettronica e telecomunicazioni. Nel CD-ROM sono raccolti i dati di oltre 200 valvole specifiche per impieghi audio.

Le Valvole Termoioniche, con la loro lunga storia, ci restituiscono ancora oggi tutto lo splendore del suono nelle sue molteplici sfumature offrendoci la possibilità di utilizzare il termine Audio nella più alta eccezione. Il CD-ROM raccoglie i dati di oltre duecento Valvole specifiche per impieghi audio, sia di produzione attuale che passata, per rispondere in tal modo alle esigenze dei progettisti quanto degli audiofili che possiedono preamplificatori, amplificatori e radioricevitori vintage. Uno strumento unico e completo che è a vostra disposizione per farvi apprezzare ancora di più le infinite possibilità applicative dei "tubi a vuoto".

CARATTERISTICHE

Oltre 200 tipi di valvole trattate, Americane, Europee ed Asiatiche. Per ogni valvola troviamo descrizione, piedinatura, equivalenti, caratteristiche statiche, caratteristiche dinamiche, eventuale schema applicativo, condizioni operative reali, curve caratteristiche.
Nelle applicazioni per amplificatore sono trattate le classi A, AB, B con schemi a valvola singola oppure push-pull. Per gli amplificatori sono forniti le tensioni e correnti, il guadagno, la distorsione, la potenza in uscita, il valore dei componenti dello schema applicativo.
Non mancano grafici che mostrano le tipiche caratteristiche statiche, dinamiche e specifiche per l'applicazione.
Infine, troviamo un glossario sempre disponibile per una facile consultazione della terminologia tecnica.
I contenuti del CD-ROM, liberamente stampabili sono interamente in italiano.
Il CD-ROM funziona su PC con sistema operativo Windows 95 o superiore.

Il CD-ROM "Valvole Audio" costa € 49,00 e si può trovare presso uno dei punti vendita Pianeta Elettronica / Eurocom-Pro. Per conoscere il punto vendita più vicino, per ulteriori informazioni sul prodotto, o per ricevere il CD-ROM direttamente a domicilio, far riferimento a:

Pianeta Elettronica
Tel. 06.454.33.731
servizio@pianetaelettronica.it
www.pianetaelettronica.it

Figura 1: *Esempio di schermata*

Scheda di richiesta abbonamento

FE - 231

Si, desidero abbonarmi a a partire dal primo numero raggiungibile

Cognome ____________________ Nome ____________________

Azienda ____________________

Via ____________________ CAP ________ Città ____________________ Prov. ________

Tel. ____________________ Fax ____________________ email ____________________

Abbonamento: *(barrare la casella prescelta)*

- ❑ **Standard:** Mi abbono a Fare Elettronica per un anno (11 uscite) a soli € 39,00 anziché € 51,00
- ❑ **Rinnovo:** Sono già abbonato ed intendo rinnovare il mio abbonamento in scadenza. Fare Elettronica per un anno (11 uscite) a soli € 39,00 anziché € 51,00, il mio codice abbonamento è ____________________
- ❑ **Regalo:** Regalo ad un amico Fare Elettronica per un anno (11 uscite) a soli € 35,00 anziché € 51,00 ***Riservato agli abbonati***, il mio codice abbonamento è ____________________
- ❑ **Scuole:** Cinque abbonamenti a Fare Elettronica per un anno (11 uscite) a soli € 156,00 anziché € 195,00 ***Riservato a Scuole ed Università***

Pagherò con: *(barrare la casella prescelta)*

- ❑ **Bollettino postale** Utilizzare il **C/C N. 22790232** intestato ad **Inware srl**, indicando nella causale **“Abbonamento a Fare Elettronica”**
- ❑ **Bonifico bancario** Appoggiarlo sulla banca: **Poste Italiane** **CIN: Z - ABI: 07601 - CAB: 01600 - C/C: 000022790232**
- ❑ **Carta di credito** VISA MasterCard Titolare: ____________________ Numero: ____________________ Scadenza: ___ / ___

Per completare l’attivazione dell’abbonamento, prego comunicare gli estremi (data e modalità prescelta) dell’avvenuto pagamento via telefono al numero (+39) 02.66504794 o via fax al numero (+39) 02.66508225

Firma ____________________

Privacy. Il trattamento dei dati, in forma automatizzata e con modalità strettamente connesse ai fini, con garanzia di riservatezza, è finalizzato all’invio del presente periodico allo scopo di informare ed aggiornare i lettori e gli operatori del settore elettronico sulle novità che il mercato propone. Potranno essere esercitati i diritti di cui all’articolo 13 della legge 675/96 (accesso, correzione, cancellazione, opposizione al trattamento, ecc.). Il titolare del trattamento dei dati è Inware srl con sede a Cormano (MI) in via Cadorna 27/31. Nel caso si tratti di copia omaggio a titolo promozionale si rende noto che i dati provengono da archivi pubblici. Resta inteso che le informazioni in ns. possesso non saranno in nessun caso cedute a terzi.

Abbonati subito!

- Compila il coupon e invialo via fax al numero 02.66508225
- Abbonati on-line: www.farelettronica.com
- Spedisci questo coupon in una busta chiusa a INWARE Edizioni Via Cadorna, 27/31 - 20032 Cormano (MI)
- Chiamaci al numero 02.66504794

INWARE EDIZIONI